MITSUBISHI

PROGRAMMABLE CONTROLLER

# MELSEC-A

Operating Manual

## Programming unit type A8PUE

JUM-406

MITSUBISHI ELECTRIC

## REVISIONS

* The manual number is given on the bottom left of the back cover.

| Print Date | *Manual Number | Revision |
|---|---|---|
| Mar., 1993 | IB (NA) 66406-A | First edition |

# INTRODUCTION

Thank you for choosing the Mitsubishi MELSEC-A Series of General Purpose Programmable Controllers. Please read this manual carefully so that the equipment is used to its optimum. A copy of this manual should be forwarded to the end User.

# CONTENTS


**1. GENERAL DESCRIPTION ..... 1 – 1 ~ 1 – 3**

1.1 Features ..... 1 – 1
1.2 Included Items ..... 1 – 3

**[BASICS]**

**2. LET'S TRY USING THE A8PUE ..... 2 – 1 ~ 2 – 20**
2.1 Before Inputting a Sequence Program ..... 2 – 3
2.1.1 Let's try connecting the A8PUE to an ACPU ..... 2 – 3
2.1.2 Let's try clearing all the memory contents of an ACPU ..... 2 – 4
2.2 Let's Try Inputting a Sequence Program ..... 2 – 5
2.2.1 Let's try writing a new sequence program ..... 2 – 5
2.2.2 Let's try reading the sequence program ..... 2 – 7
2.2.3 Let's try changing (overwriting) an instruction ..... 2 – 8
2.2.4 Let's try inserting (adding) of an instruction ..... 2 – 9
2.2.5 Let's try deleting an instruction ..... 2 – 10
2.3 Let's Try Monitoring the Sequence Program ..... 2 – 11
2.3.1 Let's try monitoring the execution status of a program ..... 2 – 12
2.4 Let's Try Testing the Sequence Program ..... 2 – 14
2.4.1 Forcibly turning a bit device ON/OFF ..... 2 – 14
2.4.2 Changing the current value of a word device ..... 2 – 16
2.4.3 Changing the set value of a timer when the CPU is in the RUN state ..... 2 – 18
2.5 Let's Try Using Help ..... 2 – 19
2.5.1 Reading an instruction in a sequence program ..... 2 – 19
2.5.2 Changing of the display format of a current value ..... 2 – 20

**[DETAILS]**

**3. SPECIFICATIONS ..... 3 – 1 ~ 3 – 7**

3.1 General Specifications ..... 3 – 1
3.2 Performance Specifications ..... 3 – 1
3.3 System Configuration ..... 3 – 2
3.3.1 System configuration ..... 3 – 2
3.3.2 Connectable ACPUs ..... 3 – 3
3.3.3 System equipment ..... 3 – 3
3.4 Lists of Functions ..... 3 – 4
3.4.1 List of functions ..... 3 – 4
3.4.2 List of help functions ..... 3 – 6

**4. HANDLING AND NOMENCLATURE ..... 4 – 1 ~ 4 – 6**

4.1 Precautions When Handling the A8PUE ..... 4 – 1
4.2 Nomenclature ..... 4 – 2
4.3 Layout of the Keyboard and List of Key Functions ..... 4 – 3

4.4 Maintenance ..... 4 – 6

**5. OPERATING PROCEDURES ..... 5 – 1 ~ 5 – 21**

5.1 Procedure From the Beginning to the End of the Operation ..... 5 – 1
5.1.1 Connecting the A8PUE to an ACPU ..... 5 – 2
5.1.2 A8PUE software version and CPU model name display ..... 5 – 4
5.1.3 Key word input ..... 5 – 5
5.1.4 Mode selection and operation ..... 5 – 7
5.1.5 Disconnecting the A8PUE from an ACPU ..... 5 – 8
5.2 How to Reset the A8PUE ..... 5 – 9
5.3 Display Adjustments and Display Format ..... 5 – 10
5.3.1 How to adjust the brightness ..... 5 – 10
5.3.2 Display backlight ON/OFF states ..... 5 – 10
5.3.3 Display format ..... 5 – 11
5.4 Basic Key Operations ..... 5 – 14
5.4.1 Valid key switching (top/bottom) ..... 5 – 14
5.4.2 How to input instructions ..... 5 – 15
5.4.3 Corrective action when a wrong key has been pressed ..... 5 – 19
5.5 Functions of the A8PUE on a MELSECNET(/B) Data Link System ..... 5 – 20

**6. HOW TO USE EACH FUNCTION ..... 6 – 1 ~ 6 – 95**

6.1 How to Understand the Operation Explanations ..... 6 – 1
6.2 WRITE Mode (W) Operations ..... 6 – 2
6.2.1 Setting a designated range in the program by using NOP (Continuous writing of NOP) ..... 6 – 2
6.2.2 Writing new programs and modifying existing programs ..... 6 – 4
6.2.3 Displaying/selecting an instruction ..... 6 – 6
6.2.4 Comment display ..... 6 – 8
6.3 READ Mode (R) Operations ..... 6 – 10
6.3.1 Reading the instruction of the step number designated in the program (Instruction read by designating the step number) ..... 6 – 10
6.3.2 Reading an instruction designated in the program (Instruction read by designating an instruction) ..... 6 – 12
6.3.3 Reading an instruction by designating a used device in the program (Instruction read by designating a device) ..... 6 – 16
6.3.4 Automatic scrolling of a program ..... 6 – 19
6.4 INSERT (I) Mode Operations ..... 6 – 21
6.4.1 Inserting an instruction in a program (Program insertion) ..... 6 – 21
6.4.2 Batch moving a program ..... 6 – 23
6.4.3 Copying a program ..... 6 – 25
6.5 DELETE (D) Mode Operations ..... 6 – 27
6.5.1 Deleting an instruction in a program (Program deletion) ..... 6 – 27
6.5.2 Deletion by designating a range ..... 6 – 29
6.5.3 Batch deletion of NOP in the program ..... 6 – 31

6.6 MONITOR (M) Mode Operations ..... 6 – 32
6.6.1 Confirming the operating state by displaying a program (List monitoring) ..... 6 – 33
6.6.2 Searching a coil from the contact (Monitoring search) ..... 6 – 36
6.6.3 Device monitoring ..... 6 – 38
6.6.4 Changing the display format of a current value ..... 6 – 42
6.6.5 Setting the offline switch YES/NO display (Offline switch display) ..... 6 – 44
6.7 TEST (T) Mode Operations ..... 6 – 46
6.7.1 Setting/resetting X, Y, M, L, B, and F by list monitoring (Set/reset of a bit device) ..... 6 – 47
6.7.2 Changing the current values of T, C, D, W, R, A, Z, or V by list monitoring (Changing current values of word devices) ..... 6 – 49
6.7.3 Setting/resetting X, Y, M, L, B, or F by device monitoring (Bit device set/reset) ..... 6 – 51
6.7.4 Changing the current value of T, C, D, W, R, A, Z, or V by device monitoring (Current value change of a word device) ..... 6 – 53
6.7.5 Setting/canceling an offline switch of Y, M, L, B, or F by device monitoring (Setting/cancellation of an offline switch) ..... 6 – 55
6.8 Parameter Setting ..... 6 – 57
6.8.1 All clearing of parameters ..... 6 – 58
6.8.2 Parameter setting ..... 6 – 60
6.9 OTHERS (O) Mode Operations ..... 6 – 67
6.9.1 Changing the T/C set values when the ACPU is in the RUN state ..... 6 – 68
6.9.2 Checking an error step/error code When an error occurs (Error check) ..... 6 – 70
6.9.3 Checking a program ..... 6 – 72
6.9.4 Monitoring the MELSECNET(II)/B link state (link monitoring) ..... 6 – 74
6.9.5 Monitoring the buffer memory of a special-function module (Buffer memory batch monitoring) ..... 6 – 79
6.9.6 Monitoring the clock data of an ACPU (clock monitoring) ..... 6 – 81
6.9.7 All clearing the memory contents of an ACPU (PC memory all clear) ..... 6 – 82
6.9.8 Clearing sequence programs, microcomputer programs, and T/C set value areas (Program all clear) ..... 6 – 83
6.9.9 Clearing the device memory of an ACPU (Device memory all clear) ..... 6 – 84
6.9.10 Setting the PC number ..... 6 – 85
6.9.11 Switching main/sub-programs ..... 6 – 86
6.9.12 Executing remote RUN/STOP ..... 6 – 87
6.9.13 Reading/writing memory contents by using machine language ..... 6 – 88
6.9.14 Setting write enabled/disabled when the CPU is in the RUN state and setting only MONITOR/TEST mode enabled (Program mode selection) ..... 6 – 91
6.9.15 Setting conductivity display YES/NO ..... 6 – 94
6.9.16 Setting the buzzer ON/OFF when a key is pressed (Buzzer setting) ..... 6 – 95

**7. LISTS OF ERROR MESSAGES ..... 7 – 1 ~ 7 – 6**

7.1 Errors Detected by the A8PUE ..... 7 – 1
7.2 PC CPU Errors ..... 7 – 4
7.3 Errors When Using the A8PUE in a Link System ..... 7 – 5

**APPENDICES** ..... **APP – 1 ~ APP – 7**

APPENDIX 1 COMPARING THE A8PUE WITH THE A7PU/A7PUS ..... APP – 1
APPENDIX 2 A8PUE OUTSIDE DIMENSIONS ..... APP – 4
APPENDIX 3 ACPU PARAMETER SETTING SHEET ..... APP – 5
APPENDIX 4 ASCII DISPLAY CHARACTER CODE ..... APP – 7

IBM, IBM-DOS(PC-DOS) are registered trademark of the International Business Machines Corporation.

## How to use this manual

This manual is structured as follows:

General description

The features of the A8PUE programming unit (hereafter called the A8PUE) are explained.

Basics

Operations are explained by using examples, so that somebody using the A8PUE for the first time can easily understand the operations.

1) Connection methods

2) Inputting, modifying, monitoring, and testing of sequence programs

Details

The details of the functions and the operating methods of the A8PUE are explained. Use this as a dictionary.

In addition, the following manuals give details about the instructions explained in this manual.

- ACPU Programming Manual (Fundamental) IB-66249
- ACPU Programming Manual (Common Instructions) IB-66250

# 1. GENERAL DESCRIPTION

This manual explains specifications, handling, and operations of the A8PUE.

The A8PUE is a peripheral device that is used with the MELSEC-A series of general-purpose programmable controllers. It can read from and write to sequence programs in a MELSEC-A series PC CPU.
The A8PUE is also used for monitoring and testing devices. Follow the procedures in this manual when using the A7PUS to perform program I/O, as well as inspection and maintenance.

## 1.1 Features

(1) A2A(S1) and A3ACPU are supported
All A2A(S1) and A3ACPU devices are supported.

(2) Expanded display area
The display area was expanded by using a back-lit LCD of 4 lines x 20 character. Therefore, inputting/editing/monitoring/testing of sequence programs can be easily executed.

(3) Easy key operations
Key operations are easier thanks to the long key stroke.

(4) Accessing another station's PC CPU is enabled
Writing, monitoring, and testing of sequence programs can be executed from the station to which A8PUE is connected to another station (master station/local station) on MELSECNET(II)/B.

(5) Writing when the CPU is in the RUN state is enabled
If the number of steps in a sequence program changes, the sequence program can be changed when the CPU is in the RUN state.

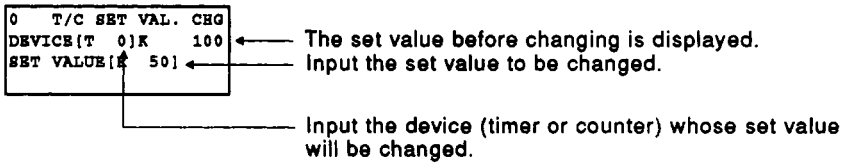
(6) Extensive help functions
Help functions for reading/writing/inserting/deleting/monitoring/testing operations in the menu selection format using dialog are supported.

(7) Easier control of parameters and sequence programs
Checking, revising, changing, inserting, and monitoring of PC CPU parameters and sequence programs is now easier.

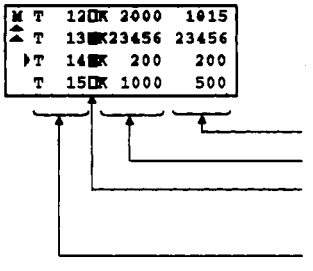
1) Example of changing a sequence program instruction

```
LD   X0          Change    LD   X0
OUT  Y20    ------------>   MOV  D0   D1
LD   X1                    LD   X1
```

2) Example of changing the T/C set value
The T/C set value can be changed in the menu format when the CPU is in the RUN state.

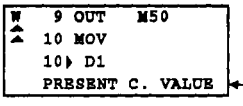

0 T/C SET VAL. CHG
DEVICE[T 0]K 100
SET VALUE[K 50]
The set value before changing is displayed.
Input the set value to be changed.
Input the device (timer or counter) whose set value will be changed.

3) Device memory monitoring example
The device memory of a PC CPU can be easily checked.

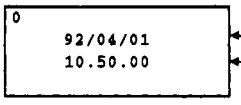



T12: Set value = 2000, Current value = 1015, Contact = OFF
T13: Set value = 23456, Current value = 23456, Contact = ON
T14: Set value = 200, Current value = 200, Contact = ON
T15: Set value = 1000, Current value = 500, Contact = OFF

(8) Comments of each device can be displayed
The comment of the device where the cursor is can be displayed by using a help function.



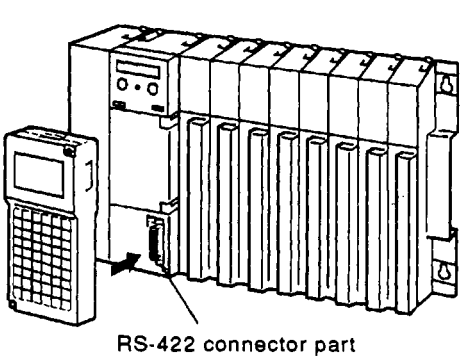

(9) Clock display is enabled
Clock data in the PC CPU can be displayed. (The A8PUE turns the clock data read request M9028 ON and OFF automatically.)

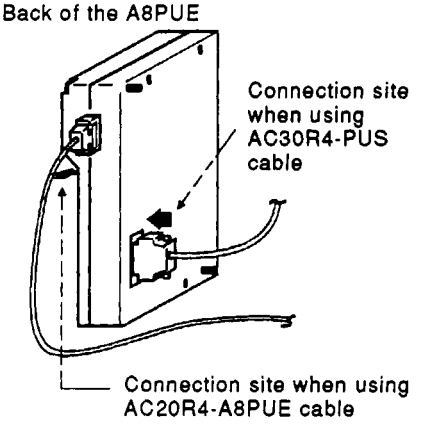


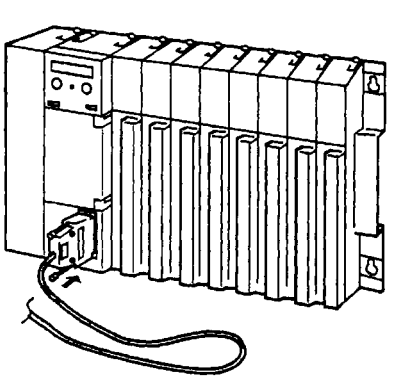

(10) Connections using the add-on or hand-held methods are enabled
There are 2 ways to connect A8PUE to a PC CPU.

[Add-on method]
Connects the A8PUE directly to a PC CPU.

[Hand-held method]
Connect the A8PUE to a PC CPU via an AC20R4-A8PU or AC30R4-PUS cable.



## 1.2 Included Items

After buying the A8PUE, make sure the following items are included:

| Model Name | Items | Amount | Remarks |
|---|---|---|---|
| A8PUE | Programming unit | 1 | ———— |
| ———— | Protective caps for the RS-422 connectors | 2 | Placed on the connectors at the factory |

The following abbreviations are used in this manual:

(1) A8PUE
A8PUE Programming Unit

(2) ACPU
A-series PC CPUs to which the A8PUE can be connected (see Section 3.3.2).

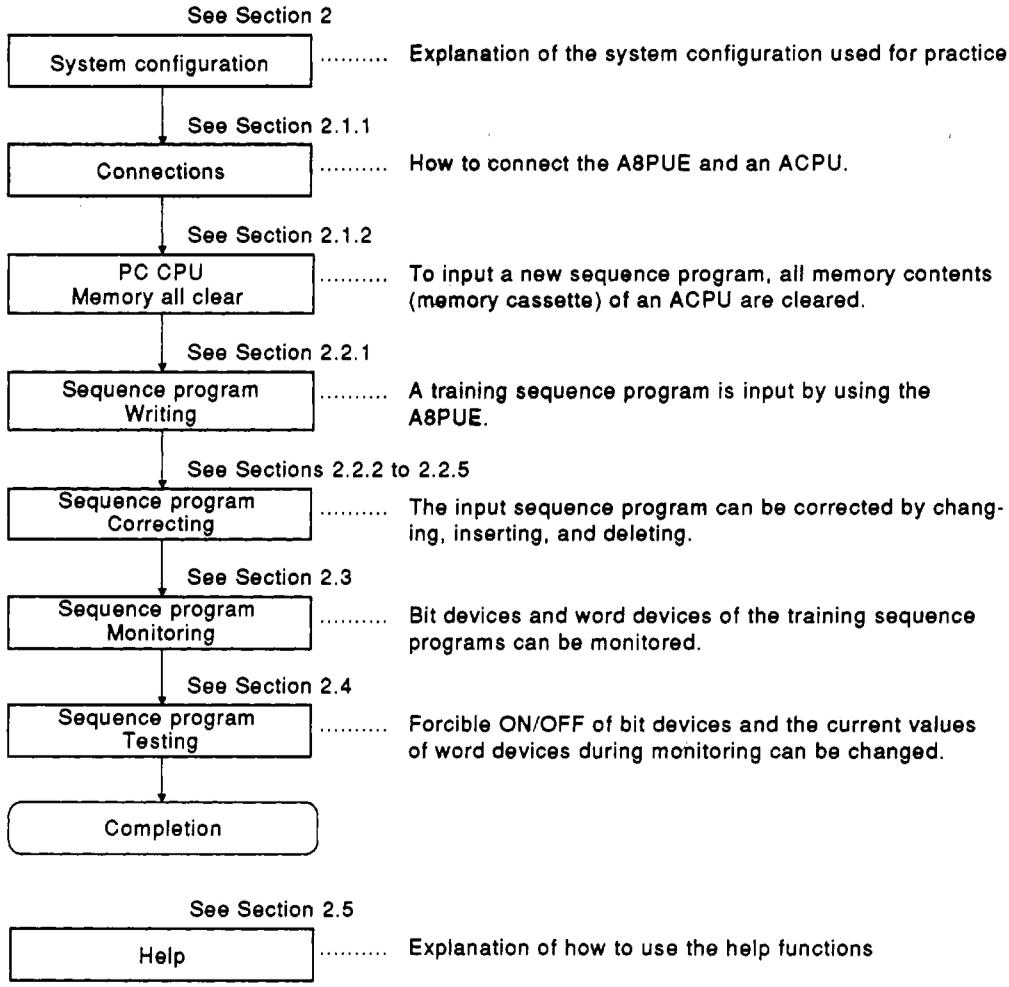
(3) [ ] [ ] [ ] [ ] CPU
Including PC CPUs with the MELSECNET(II) link function.
(Also including A1CPU → A1CPUP21/R21.)

(4) Peripheral devices
All peripheral devices equipped with usable GPP functions in the MELSEC-A series

# [BASICS]

# 2. LET'S TRY USING THE A8PUE

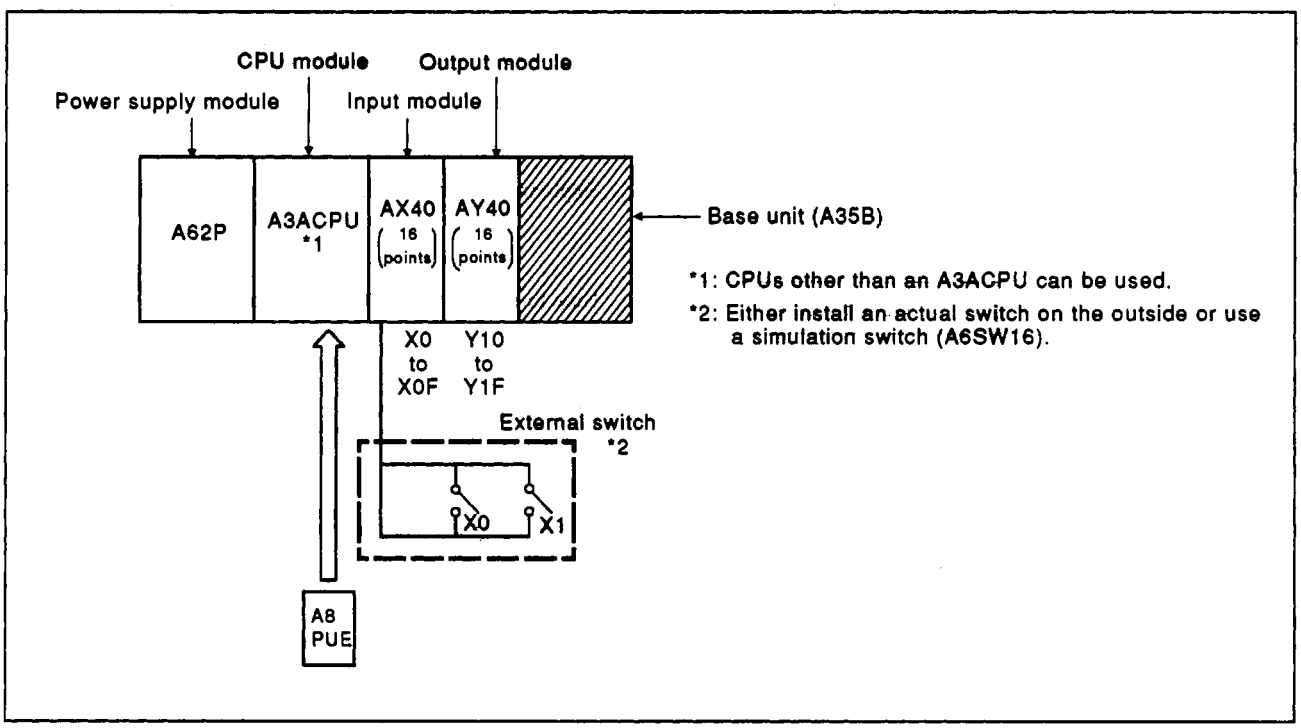
It is possible to understand A8PUE operations quickly by practicing the creation and monitoring of actual programs. A very basic training program can be created and its operations can be checked by using the monitoring test function.
A simple example is shown below. Try to operate the A8PUE in accordance with the example.


See Section 2
System configuration ..... Explanation of the system configuration used for practice
See Section 2.1.1
Connections ..... How to connect the A8PUE and an ACPU.
See Section 2.1.2
PC CPU Memory all clear ..... To input a new sequence program, all memory contents (memory cassette) of an ACPU are cleared.
See Section 2.2.1
Sequence program Writing ..... A training sequence program is input by using the A8PUE.
See Sections 2.2.2 to 2.2.5
Sequence program Correcting ..... The input sequence program can be corrected by changing, inserting, and deleting.
See Section 2.3
Sequence program Monitoring ..... Bit devices and word devices of the training sequence programs can be monitored.
See Section 2.4
Sequence program Testing ..... Forcible ON/OFF of bit devices and the current values of word devices during monitoring can be changed.
Completion
See Section 2.5
Help ..... Explanation of how to use the help functions

Since this section explains only the basic operations of the A8PUE, see Section 6 for greater details about operations and the contents of each mode.

The system configuration to actually perform basic operations is shown below.

**[System configuration]**

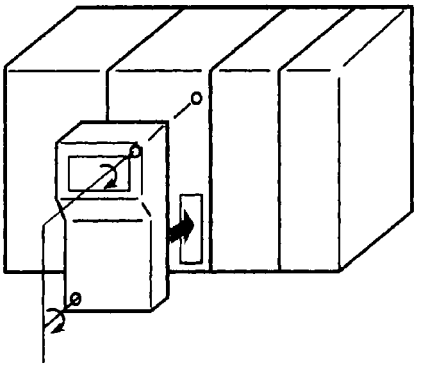


## 2.1 Before Inputting a Sequence Program

This section explains how to connect the A8PUE to an ACPU and how to clear the entire memory (memory all clear) to write a new sequence program to the ACPU.

### 2.1.1 Let's try connecting the A8PUE to an ACPU

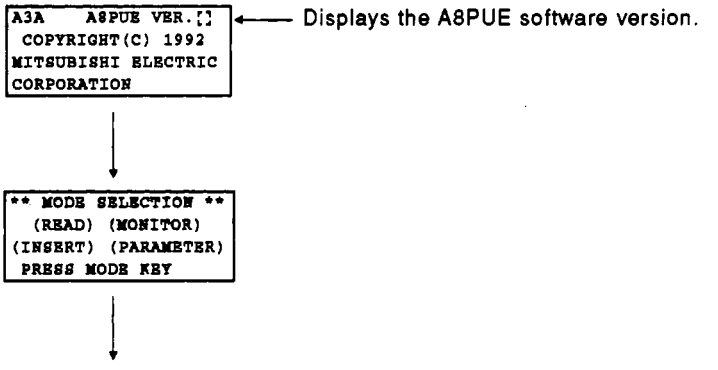
1) Connect the A8PUE to an ACPU as follows:



Tighten the anchor screws of the A8PUE.

2) Put the ACPU in the STOP state.

3) Turn ON the power supply.
The display area of A8PUE can be switched as shown below.




Then, key operations in all modes can be done.

Let's try executing a memory all clear in the CPU so that we can write a new sequence program to the ACPU.

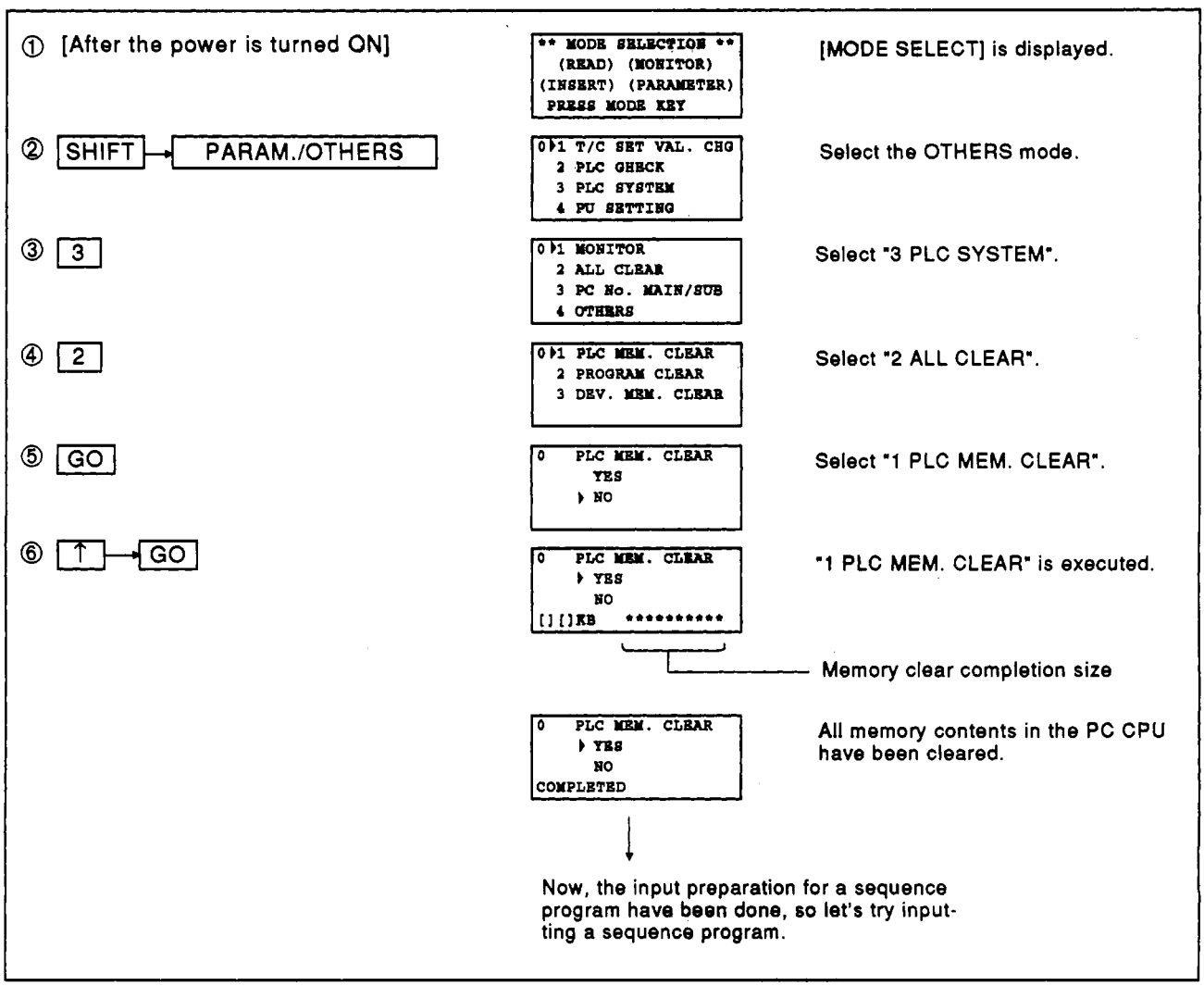
**REMARK**

When a key word is set, operate as shown in Section 5.1.3.

### 2.1.2 Let's try clearing all the memory contents of an ACPU

This section shows how to clear all memory contents (memory cassette) to write a new sequence program to an ACPU.

**[Sample operation]**



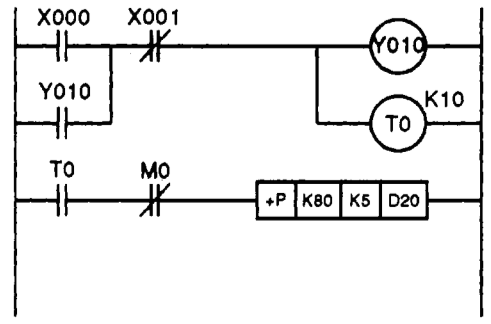
## 2.2 Let's Try Inputting a Sequence Program

This section explains how to write, read, change, insert in, and delete sequence programs.

### 2.2.1 Let's try writing a new sequence program

Input the following sequence program:

```
 0 LD   X000
 1 OR   Y010
 2 ANI  X001
 3 OUT  Y010
 4 OUT  T0   K10
 5 LD   T0
 6 ANI  M0
 7 +P   K80  K5   D20
14 END
```




Step Number: Displayed automatically when input

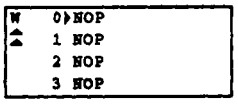
How to correct a wrong input:
Either see the operations in Sections 2.2.2 to 2.2.5 or execute a memory all clear (see Section 2.1.2) and input from the beginning.

**[Sample operation]**

① SHIFT → READ/WRITE

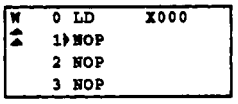
Select the WRITE mode.

② STEP NUMBER → 0 → GO

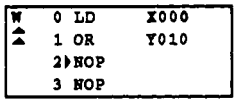



Read step 0.

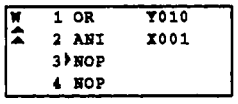
③ LD → X → 0 → GO



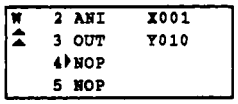

④ OR → Y → 1 → 0 → GO



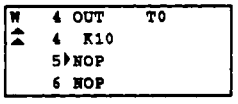

⑤ ANI → X → 1 → GO



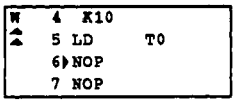

⑥ OUT → Y → 1 → 0 → GO



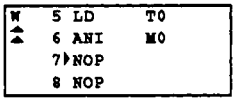

⑦ OUT → T → 0 → SP → K → 1 → 0 → GO



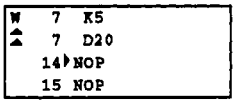

⑧ LD → T → 0 → GO




⑨ ANI → M → 0 → GO

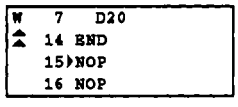



⑩ + → P → K → 8 → 0 → SP → K → 5 → SP → D* → 2 → 0 → GO




* : Both the D and AND D keys are valid.

⑪ END → GO

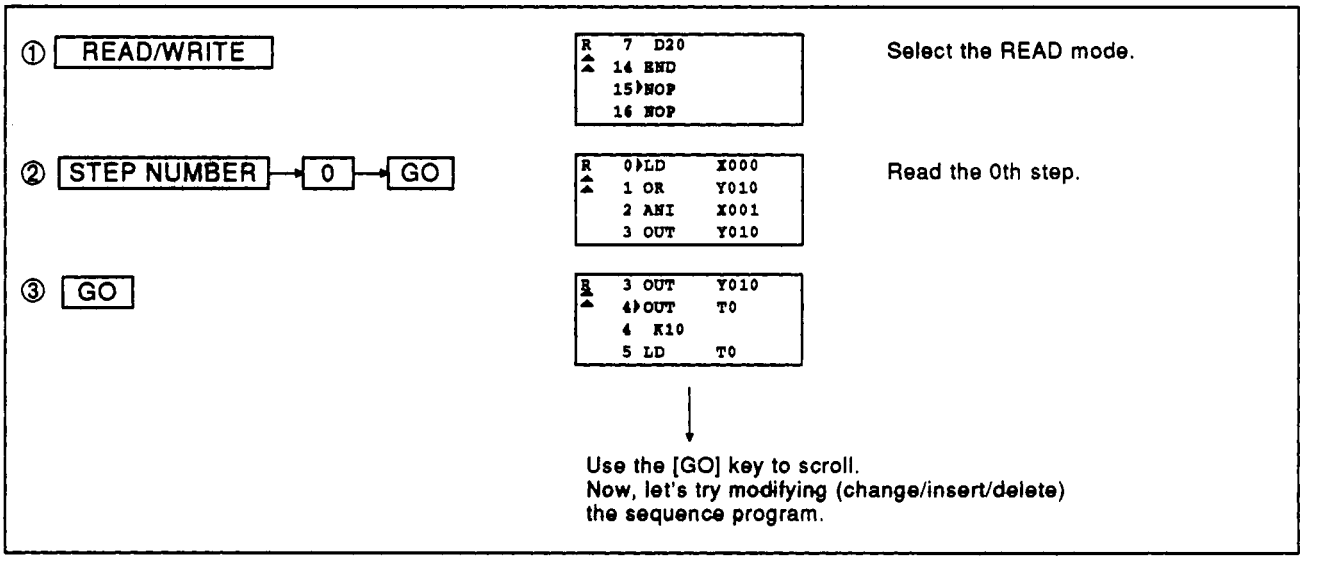



The sequence program has now been input.
Now, let's try reading the input sequence program from a PC CPU to the A8PUE.

### 2.2.2 Let's try reading the sequence program

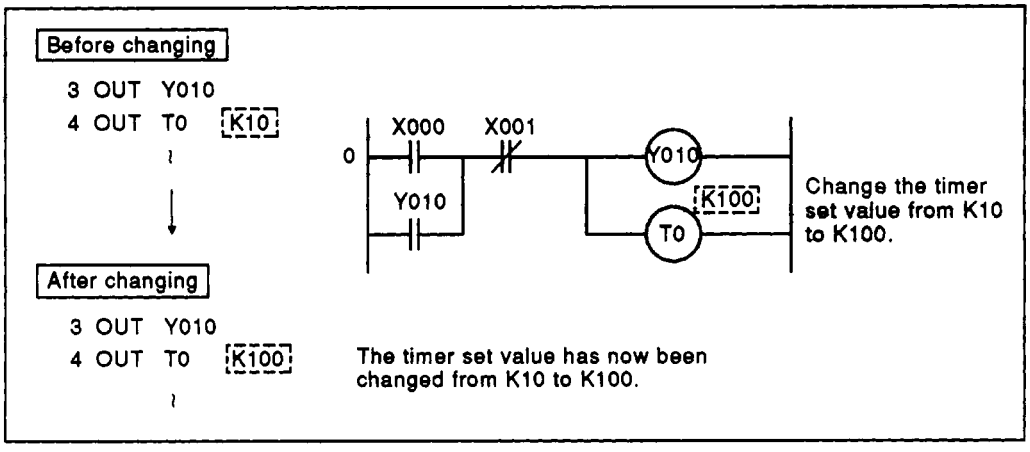
This section shows how to read the sequence program written in Section 2.2.1 to check it.

**[Sample operation]**

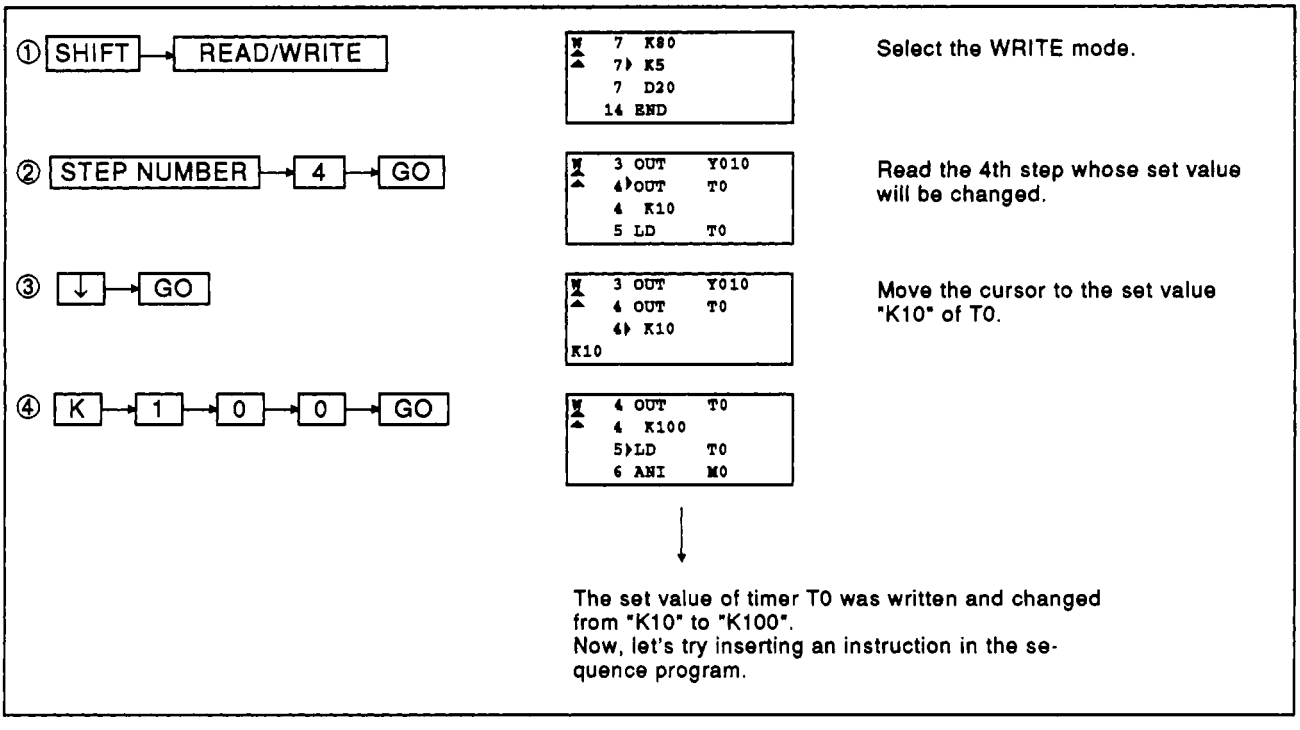


### 2.2.3 Let's try changing (overwriting) an instruction

This section shows how to change the sequence program written in Section 2.2.1.

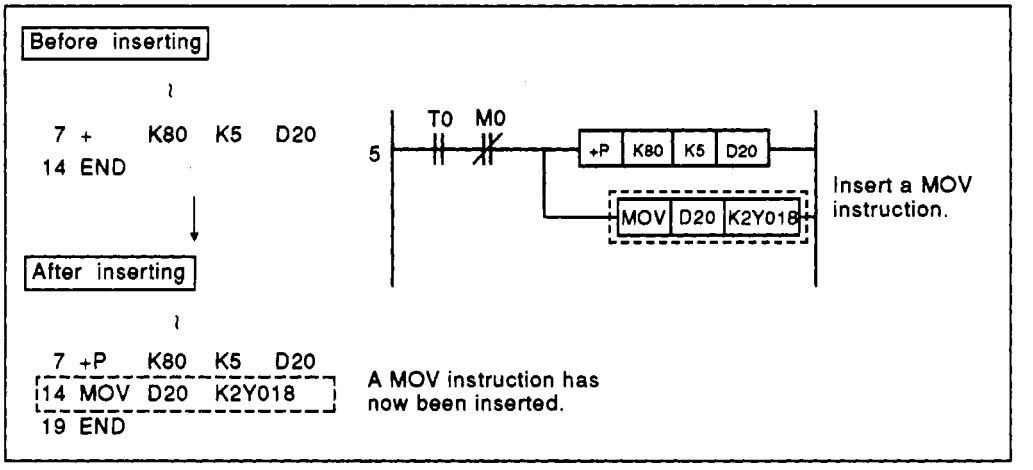



**[Sample operation]**

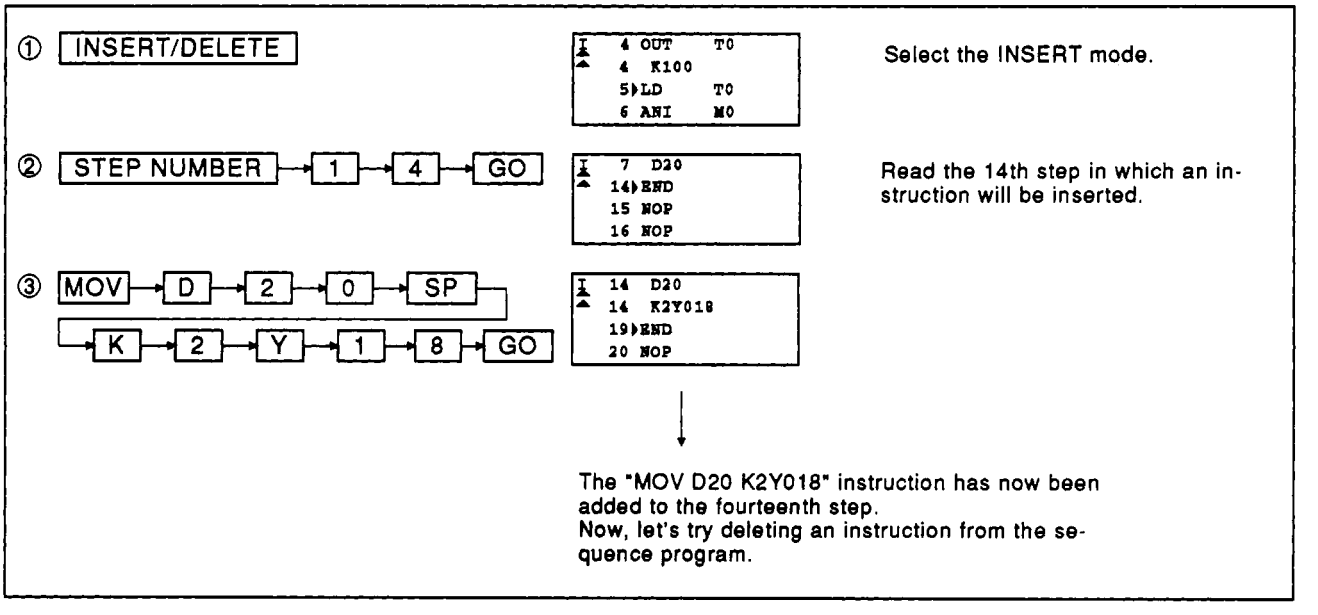

① SHIFT → READ/WRITE
② STEP NUMBER → 4 → GO
③ ↓ → GO
④ K → 1 → 0 → 0 → GO
W
7 K80
7 K5
7 D20
14 END
3 OUT Y010
4 OUT T0
4 K10
5 LD T0
3 OUT Y010
4 OUT T0
4 K10
K10
4 OUT T0
4 K100
5 LD T0
6 ANI M0
Select the WRITE mode.
Read the 4th step whose set value will be changed.
Move the cursor to the set value "K10" of T0.
The set value of timer T0 was written and changed from "K10" to "K100".
Now, let's try inserting an instruction in the sequence program.

### 2.2.4 Let's try inserting (adding) of an instruction

This section shows how to insert an instruction to the program input in Section 2.2.1.

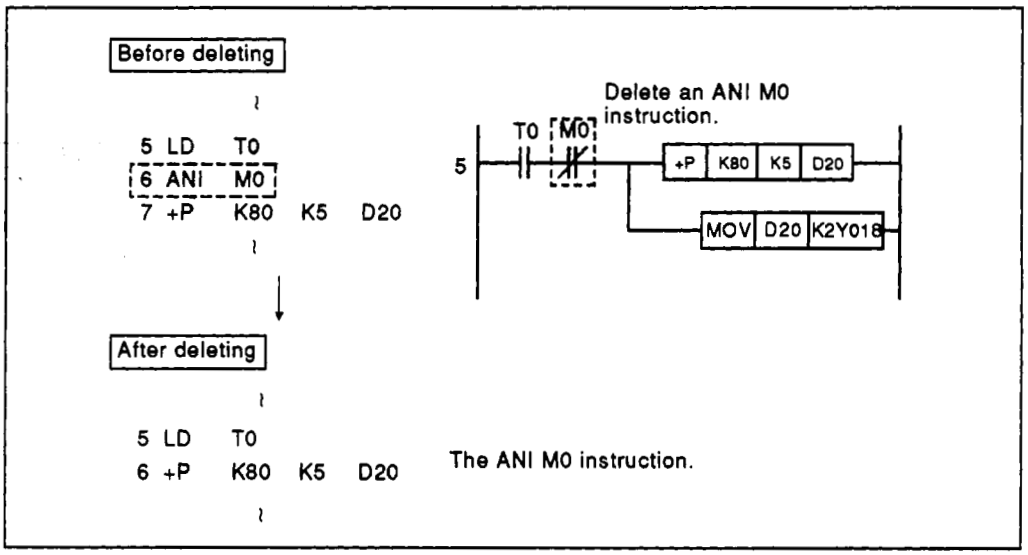



**[Sample operation]**


① INSERT/DELETE
4 OUT T0
4 K100
5 LD T0
6 ANI M0
Select the INSERT mode.
② STEP NUMBER 1 4 GO
7 D20
14 END
15 NOP
16 NOP
Read the 14th step in which an instruction will be inserted.
③ MOV D 2 0 SP
K 2 Y 1 8 GO
14 D20
14 K2Y018
19 END
20 NOP
The "MOV D20 K2Y018" instruction has now been added to the fourteenth step.
Now, let's try deleting an instruction from the sequence program.

### 2.2.5 Let's try deleting an instruction

This section shows how to delete an instruction from the program written in Section 2.2.1.




**[Sample operation]**

① SHIFT → INSERT/DELETE

```
R  14  D20
   14  K2Y018
   19 END
   20 NOP
```

Select the DELETE mode.

② STEP NUMBER → 6 → GO

```
R   5 LD    T0
    6 ANI   M0
    7 +P
    7  K80
```

Read the 6th step from which an instruction will be deleted.

③ GO

```
R   5 LD    T0
    6 +P
    6  K80
    6  K5
```

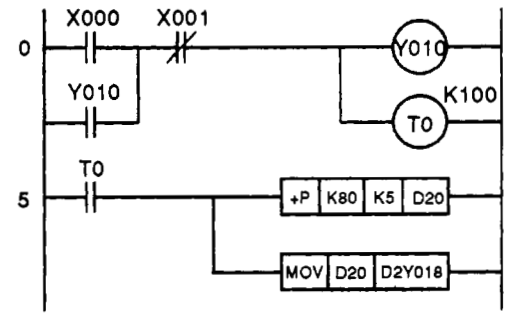
The cursor position instruction is deleted by using the [GO] key and step numbers move up accordingly.

The "ANI M0" instruction has now been deleted.
Now, let's try executing the sequence program and monitoring it by using the A8PUE.

## 2.3 Let's Try Monitoring the Sequence Program

The sequence program written in Section 2.2.1 and modified in Sections 2.2.3 to 2.2.5 is shown below.
This section shows how to execute the sequence program and monitor a bit device and a word device.

```
 0 LD    X000
 1 OR    Y010
 2 ANI   X001
 3 OUT   Y010
 4 OUT   T0    K100
 5 LD    T0
 6 +     K80   K5   D20
13 MOV   D20   K2Y018
18 END
```



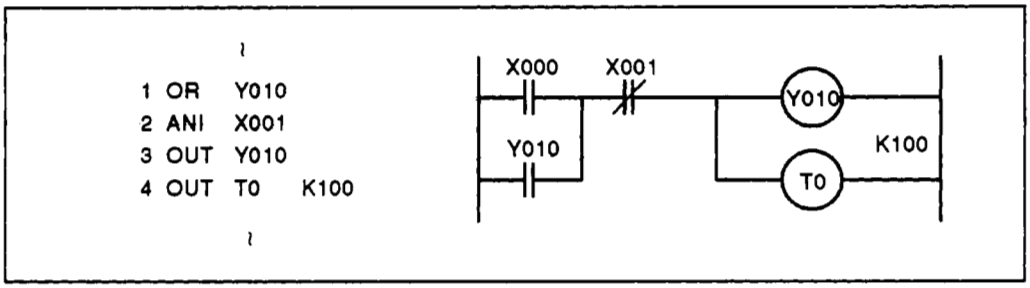

**[General description of the operations]**

The program is as follows:
Timer T0 goes ON when X0 is turned ON. And then, Y18, Y1A, Y1C, and Y1E go ON after 10 sec. (The Y18, Y1A, Y1C, and Y1E LEDs of the AY40's output module all light.)

Now, let's try executing and monitoring the sequence program written in Section 2.3.1.

### 2.3.1 Let's try monitoring the execution status of a program

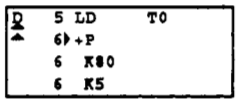
This section shows how to monitor the execution state of a sequence program, and how to check the current values of the ON/OFF states of bit devices and word devices.



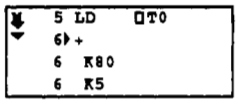

**[Sample operation]**

(1) Monitoring operation by using step number designation

① Set the RUN keyswitch to RUN.

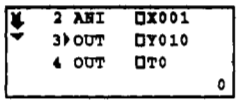



② MON./TEST

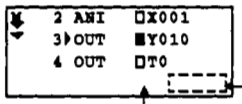



Select the MONITOR mode.

③ STEP NUMBER → 3 → GO



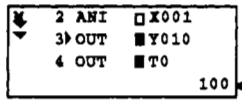

④ Turn ON the X0 external switch.




The current value of a timer is displayed.

■ : A bit device goes ON.
□ : A bit device goes OFF.

⑤ [About 10 sec. after X0 is turned ON]

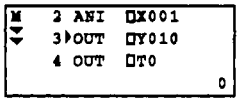



After the timer times out, the current value becomes "100".

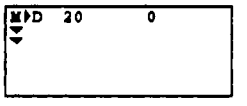
The monitoring operation has now been checked by using step number designation. Set the RUN key-switch of the CPU to STOP and RESET.
Turn OFF the X0 external switch.
Now, let's try checking the monitoring operation by using device designation.

(2) Checking the monitoring operation by using device designation
Monitoring device: D20

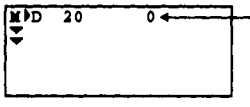
① Set the RUN keyswitch to RUN.
MON./TEST




② D → 2 → 0 → GO

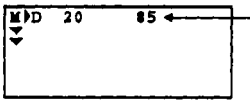



③ Turn ON the X0 external switch.




The current value of a word device is displayed.

④ [About 10 sec. after X0 is turned ON]




"85" is stored in D20.

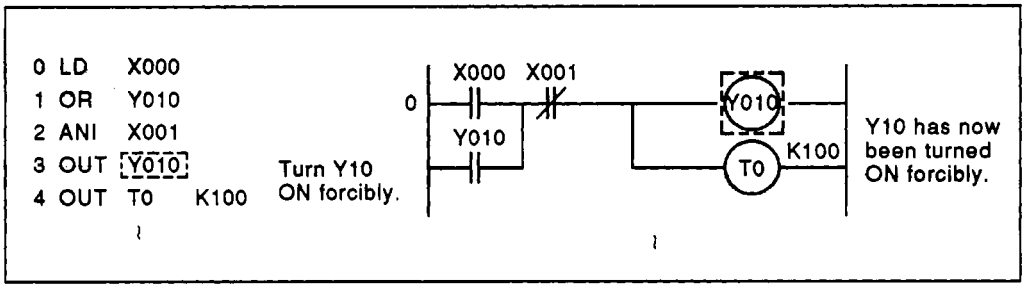
The monitoring operation has now been checked by using device designation.
Turn OFF the X0 external switch.
Now, let's try testing a device in the sequence program by using the A8PUE.

## 2.4 Let's Try Testing the Sequence Program

This section shows how to execute and test the sequence program written in Section 2.2.1 and modified in Sections 2.2.3 to 2.2.5.

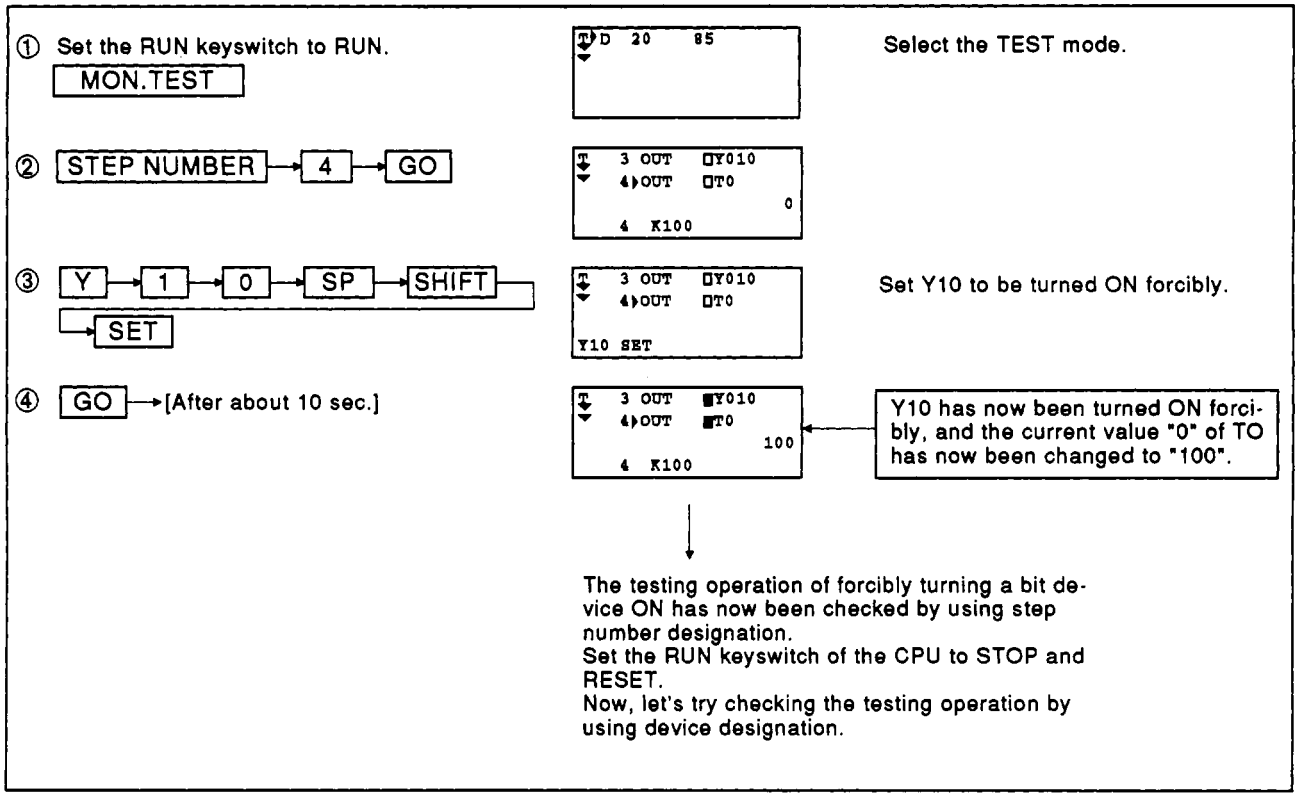
### 2.4.1 Forcibly turning a bit device ON/OFF

This section shows how to check that Y10 was turned ON forcibly, and that the current value of T0 became "100" after 10 sec.




**[Sample operation]**

(1) Testing operation by using step number designation (list monitoring)


① Set the RUN keyswitch to RUN.
MON.TEST
② STEP NUMBER → 4 → GO
③ Y → 1 → 0 → SP → SHIFT → SET
④ GO → [After about 10 sec.]
Select the TEST mode.
Set Y10 to be turned ON forcibly.
Y10 has now been turned ON forcibly, and the current value "0" of TO has now been changed to "100".
The testing operation of forcibly turning a bit device ON has now been checked by using step number designation.
Set the RUN keyswitch of the CPU to STOP and RESET.
Now, let's try checking the testing operation by using device designation.

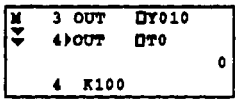
(2) Checking the testing operation by using device designation (device monitoring)

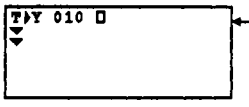
① Set the RUN keyswitch to RUN.
MON./TEST

② Y → 1 → 0 → GO → SHIFT
MON./TEST

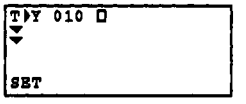
③ SHIFT → SET

④ GO

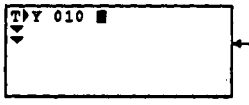



Turn OFF Y10. (□)




Set Y10 ON forcibly.

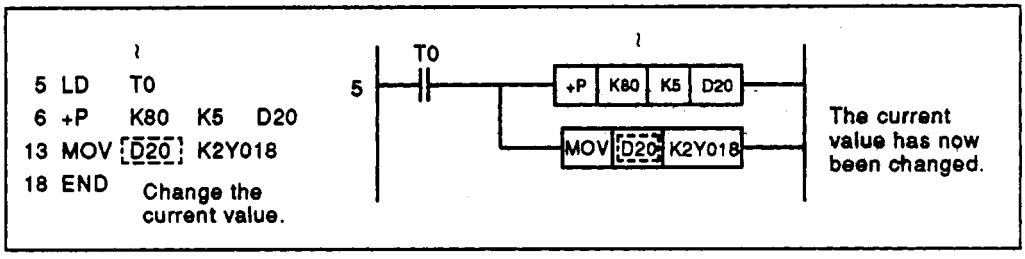



Y10 has now been turned ON forcibly.

The testing operation of forcibly turning a bit device ON by using device designation has now been checked.
Keep the CPU in the RUN state.
The following checks testing operation to which the current value of a word device is changed.

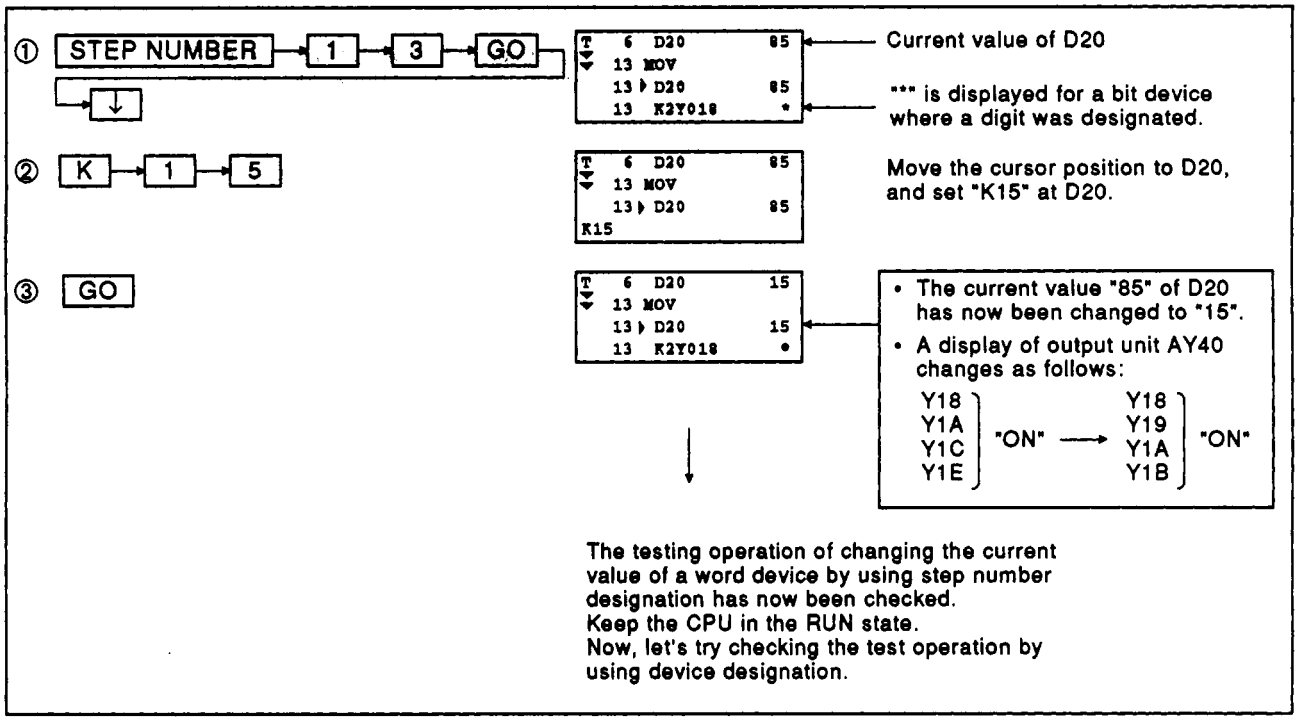
### 2.4.2 Changing the current value of a word device

This section shows how to perform the test which changes the current value of D20 when the CPU is in the RUN state.

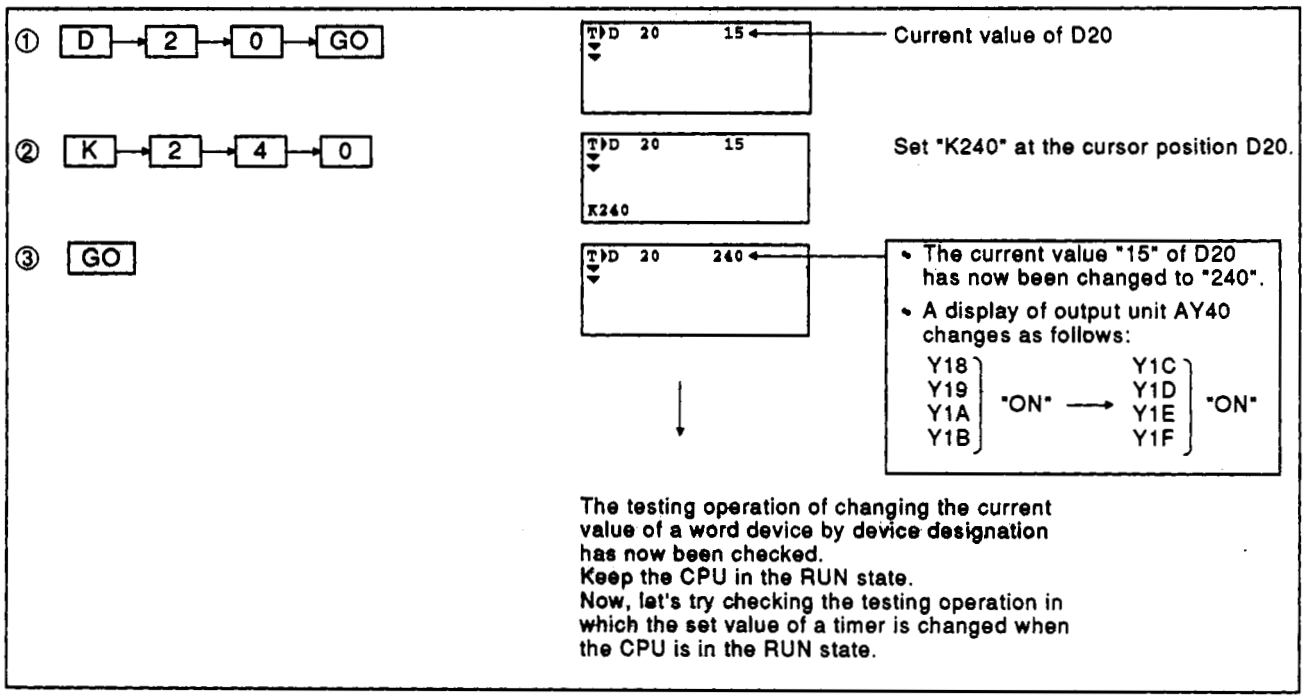



**[Sample operation]**

(1) Testing operation by using step number designation (list monitoring)
(Change the current value "85" of D20 to "15".)


① STEP NUMBER 1 3 GO
② K 1 5
③ GO
T 6 D20 85
13 MOV
13 D20 85
13 K2Y018 *
Current value of D20
*** is displayed for a bit device where a digit was designated.
T 6 D20 85
13 MOV
13 D20 85
K15
Move the cursor position to D20, and set "K15" at D20.
T 6 D20 15
13 MOV
13 D20 15
13 K2Y018 *
• The current value "85" of D20 has now been changed to "15".
• A display of output unit AY40 changes as follows:
Y18 Y1A Y1C Y1E "ON" → Y18 Y19 Y1A Y1B "ON"
The testing operation of changing the current value of a word device by using step number designation has now been checked.
Keep the CPU in the RUN state.
Now, let's try checking the test operation by using device designation.

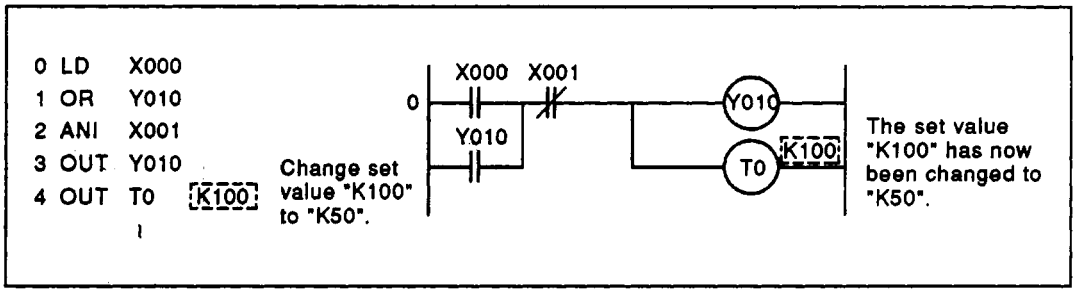
(2) Checking the test operation by using device designation (device monitoring)
(Change the current value "15" of D20 to "240".)




The testing operation of changing the current value of a word device by device designation has now been checked.
Keep the CPU in the RUN state.
Now, let's try checking the testing operation in which the set value of a timer is changed when the CPU is in the RUN state.

### 2.4.3 Changing the set value of a timer when the CPU is in the RUN state

This section shows how to forcibly change the set value of a timer of sequence programs in the RUN state.
Change the set value "K100" of timer "T0" in the program to "K50" when the CPU is in the RUN state.




**[Sample operation]**

① SHIFT → PARAM./OTHERS

```
0▶1 T/C SET VAL. CHG
  2 PLC CHECK
  3 PLC SYSTEM
  4 PU SETTING
```

Select the OTHERS mode.
Switched to the menu display.

② GO

```
0   T/C SET VAL. CHG
DEVICE[     ]
SET VALUE[      ]
```

③ T → 0 → GO

```
0   T/C SET VAL. CHG
DEVICE[T  0]K    100
SET VALUE[K   100]
```

④ 5 → 0

```
0   T/C SET VAL. CHG
DEVICE[T  0]K    100
SET VALUE[K    50]
```

⑤ GO

```
0   T/C SET VAL. CHG
DEVICE[T  0]K     50
SET VALUE[K    50]
```

Change a set value to "50".

⑥ END

```
0▶1 T/C SET VAL. CHG
  2 PLC CHECK
  3 PLC SYSTEM
  4 PU SETTING
```

The set value of a timer could be changed.
Check that the set value of timer "T0" has now been changed into "K50"(see the operation in Section 2.2.2).
This completes the basic operations of the A8PUE.
Now, let's try using the help functions.
The basic operations of the help function are explained in Section 2.5.

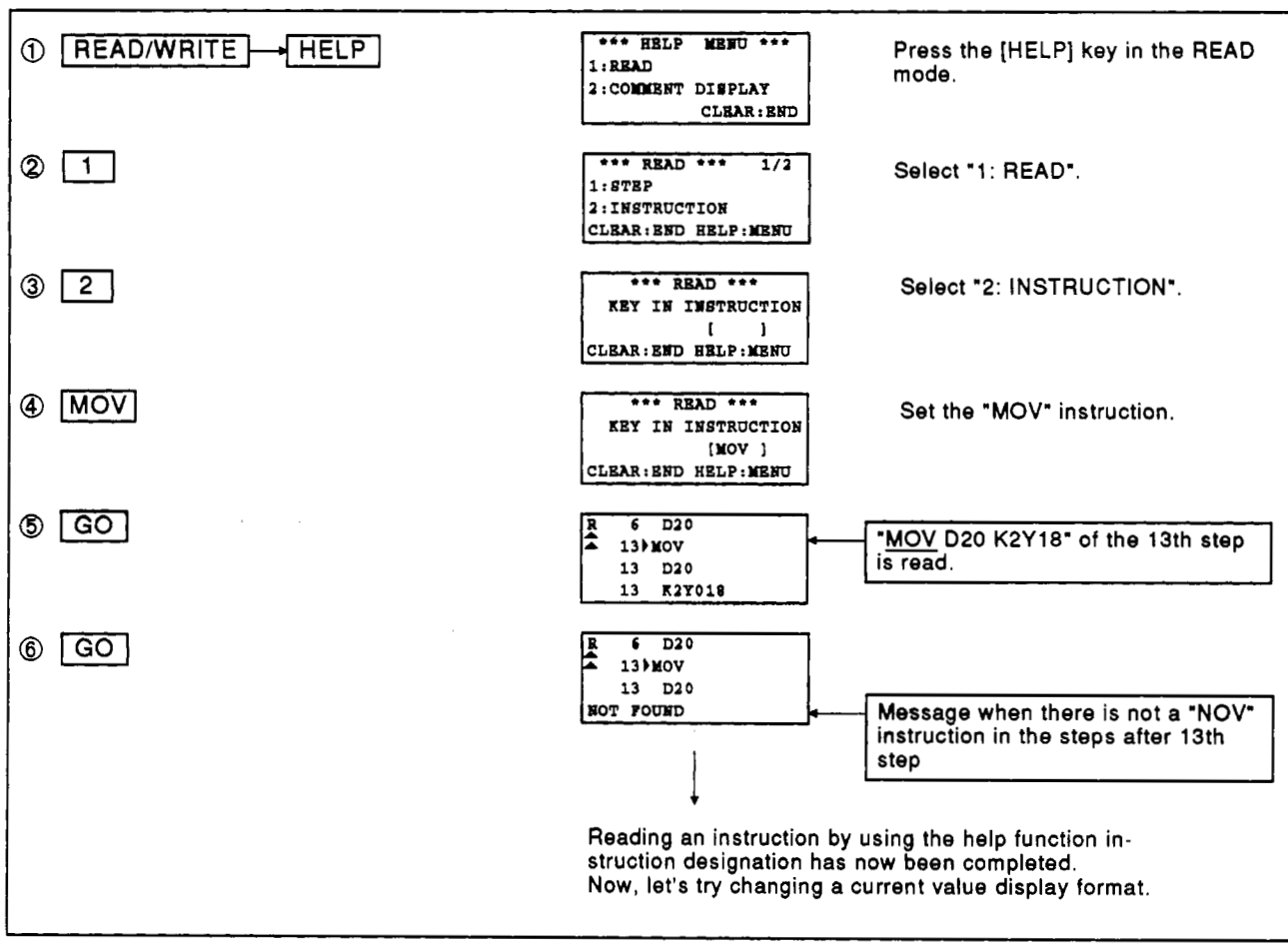
## 2.5 Let's Try Using Help

The [HELP] key is pressed when using the help functions.
The help function items of each mode are displayed in a menu format by pressing the [HELP] key. Therefore, corresponding items can be selected and executed.
This section explains, as sample operations, the reading operation by using instruction designation and the changing operation of the display format of a current value.

### 2.5.1 Reading an instruction in a sequence program

This section explains how to read the place where the "MOV" instruction is used in the sequence program.

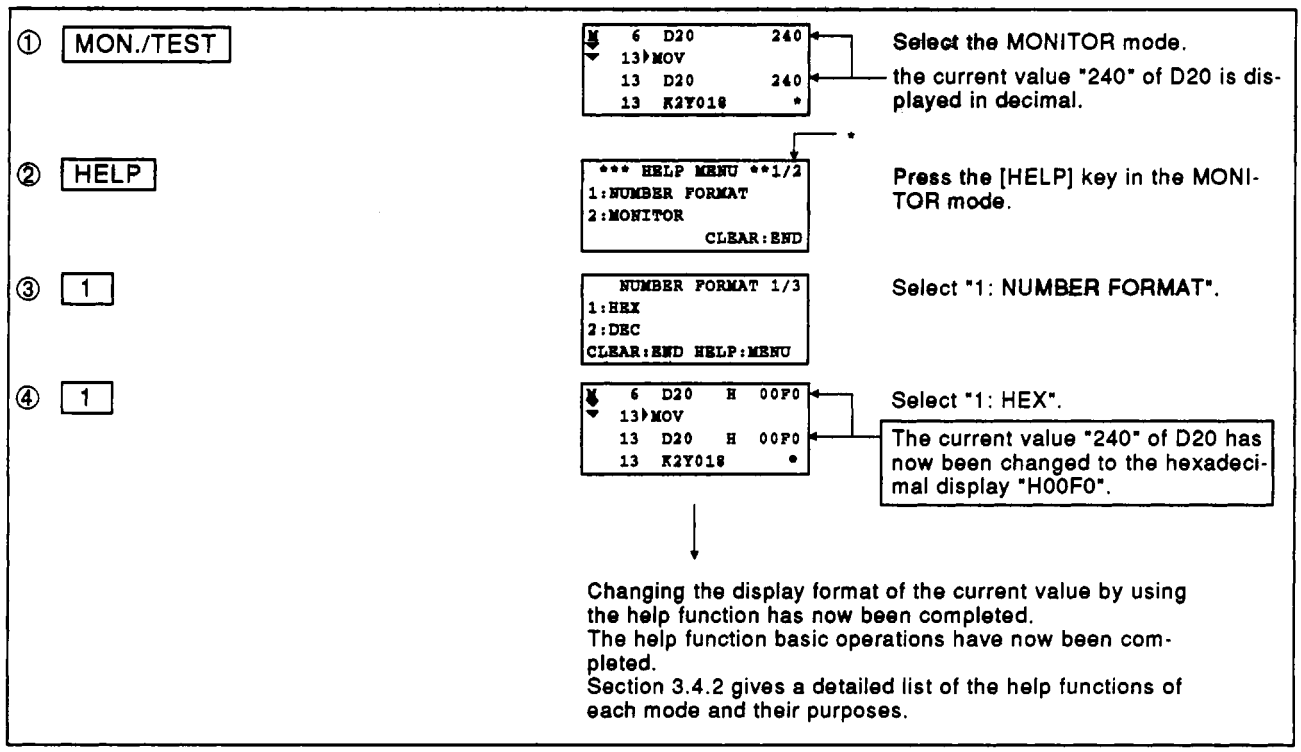
**[Sample operation]**



### 2.5.2 Changing of the display format of a current value

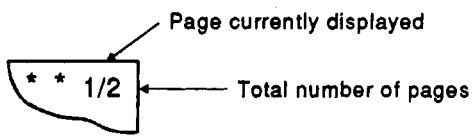
This section shows how to change the current value display format in the MONITOR mode from decimal to hexadecimal display.

**[Sample operation]**


① MON./TEST
6 D20 240
13 MOV
13 D20 240
13 K2Y018 *
Select the MONITOR mode.
the current value "240" of D20 is displayed in decimal.
② HELP
*** HELP MENU **1/2
1:NUMBER FORMAT
2:MONITOR
CLEAR:END
Press the [HELP] key in the MONITOR mode.
③ 1
NUMBER FORMAT 1/3
1:HEX
2:DEC
CLEAR:END HELP:MENU
Select "1: NUMBER FORMAT".
④ 1
6 D20 H 00F0
13 MOV
13 D20 H 00F0
13 K2Y018 *
Select "1: HEX".
The current value "240" of D20 has now been changed to the hexadecimal display "H00F0".
Changing the display format of the current value by using the help function has now been completed.
The help function basic operations have now been completed.
Section 3.4.2 gives a detailed list of the help functions of each mode and their purposes.

* When the number of pages is displayed on the upper right hand side of a help screen, press the [↑] and [↓] keys to change the pages on the screen.

Page display example



[DETAILS]

# 3. SPECIFICATIONS

## 3.1 General Specifications

**Table 3.1 A8PUE General Specifications**

| Items | Specifications | | | | | |
|---|---|---|---|---|---|---|
| Ambient temperature | Operating | 0 to 40 °C | | | | |
| | Storage | −20 to 70 °C | | | | |
| Ambient humidity | Operating | 20 to 85 % RH, no condensation | | | | |
| | Storage | 10 to 90 % RH, no condensation | | | | |
| Vibration resistance | Conforms to *JIS-C0911 | | Frequency | Acceleration | Amplitude | Sweep Count |
| | | | 10 to 55 Hz | —— | 0.075 mm (0.003 inches) | 10 times (1 octave/minute) |
| | | | 55 to 150 Hz | 9.8 $m/s^2$ (1G) | —— | |
| Shock resistance | Conforms to * JIS-C0912 (10 g x 3 times in 3 directions) | | | | | |
| Operating ambience | No corrosive gases or dust. | | | | | |
| Cooling method | Self-cooling | | | | | |

*JIS: Japanese Industrial Standard

## 3.2 Performance Specifications

**Table 3.2 A8PUE Performance Specifications**

| Items | Specifications | | |
|---|---|---|---|
| Connected module | ACPU (see Section 3.3.2) | | |
| Power, current consumption | Power supplied from connected ACPU (5 VDC, 0.4 A) | | |
| Connection method | Add-on | Attached directly to the ACPU. | |
| | Hand-held | Connected via RS-422 cable. | |
| LCD display | Display of 4 lines x 20 characters (with cursor) | | |
| Operating method | Consists of 54 operation keys (covered with polyurethane film) | | |
| Key operation check | Buzzer | | |
| Display lifespan | 100000 hours or more (when using the unit at 15 to 35 °C ambient temperature and 65 % RH or less ambient humidity) | | |
| Backlight lifespan | 50000 hours or more (when using the unit at 25 °C operating ambient temperature)<br>If ON, goes OFF if a key has not been input for 10 minutes. | | |
| Keypad lifespan | 1000000 times | | |
| External interface | RS-422 | Rear side of the unit | Add-on connection and AC30R4-PUS cable connections |
| | | Right side of the unit | AC20R4-A8PUE cable connections |
| | Extended interface | Upper side of the unit | Unused |
| Outside dimensions mm(in.) | 188(7.40) (height) x 95(3.74) (width) x 44.5(1.75) (depth)<br>When installed onto an ACPU, the depth is 37.5(1.48). | | |
| Weight kg(lb) | 0.5(1.1) | | |

## 3.3 System Configuration

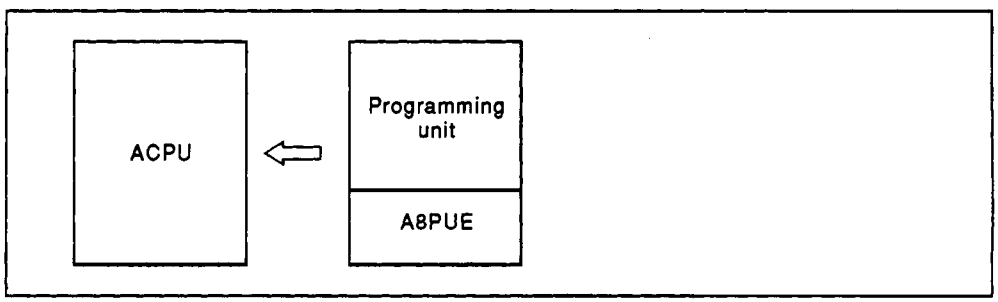
This section shows the system configuration and system equipment when using the A8PUE.

### 3.3.1 System configuration

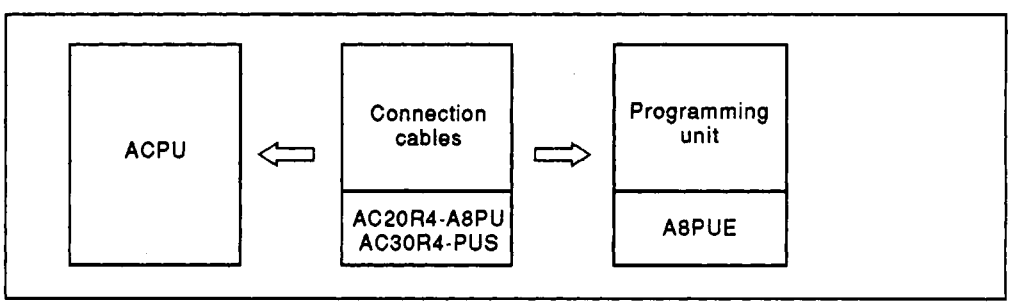
The A8PUE is connected to and operates with an ACPU by using either of the following methods:

| Connecting Methods | How to Connect |
|---|---|
| Add-on method | Attach the A8PUE directly to the ACPU. |
| Hand-held method | Connect the A8PUE and ACPU via RS-422 cable. |

(1) Add-on method




(2) Hand-held method




**REMARK**

Section 3.3.2 gives connectable ACPUs.

### 3.3.2 Connectable ACPUs

ACPUs to which the A8PUE can be connected are as follows:

| Connectable ACPUs | Connecting Methods | | |
|---|---|---|---|
| | Add-On | Hand-held | |
| | | AC20R4-A8PUE Cables | AC30R4-PUS Cables |
| A0J2CPU<br>A0J2HCPU<br>A1CPU, A1NCPU<br>A1SCPU<br>A2CPU(-S1), A2NCPU(-S1)<br>A2CCPU(C24/PRF)<br>A2ACPU(-S1)<br>A3CPU, A3NCPU<br>A3ACPU<br>A3HCPU<br>A3MCPU<br>A3VCPU<br>A73CPU, A373CPU | Applicable to other than an A1S and A2C(C24/PRF) | Applicable | Applicable |

**POINT**

The A8PUE uses power from the connected ACPU.
Since the current consumption of A8PUE is 5 VDC and 0.4 A, the power supply and the capacity of the total internal current consumption of a connected ACPU must be taken into consideration when connecting the A8PUE.

### 3.3.3 System equipment

The following table shows the equipment needed to use the A8PUE.

| Model Names | Parts | Remarks | Weights (kg) |
|---|---|---|---|
| A8PUE | Programming unit | Programming unit with an LCD and programming function | 0.5 |
| AC20R4-A8PU | RS-422 cable | Cable for connecting the RS-422 connector on the right side of the A8PUE to an ACPU<br>Length: 2 m (Option) | 0.2 |
| AC30R4-PUS | RS-422 cable | Cable for connecting the RS-422 connector on the rear side of A8PUE to an ACPU<br>Length: 3 m (Option) | 0.5 |

## 3.4 Lists of Functions

### 3.4.1 List of functions

| Modes (Mode Displays) | Functions | | | | Purposes | Reference Sections |
|---|---|---|---|---|---|---|
| WRITE (W) | Writing programs | | | | To write, insert, and change programs | 6.2.2 |
| | Device change | | | | To change devices used in designated steps in programs | 6.2.2 |
| | Help | Write | Instruction help | Instruction display /selection | Display/selection of instructions that begin with the designated characters | 6.2.3 |
| | | | | Step read | To designate step numbers and read programs | —— |
| | | | NOP continuation | | To make designated ranges in programs with NOP | 6.2.1 |
| | | Comment display | | | To display comments of designated devices | 6.2.4 |
| READ (R) | Reading programs | | | | To designate step numbers and read programs | 6.3.1 |
| | | | | | To designate utilized instructions and read programs | 6.3.2 |
| | | | | | To designate utilizd devices and read programs | 6.3.3 |
| | Automatic scrolling | | | | To display programs read to designated steps by scrolling automatically | 6.3.4 |
| | Help | Read | Step | | Same as the purposes of the read and automatic scrolling functions. | —— |
| | | | Instruction | | | —— |
| | | | Device | | | —— |
| | | | Automatic scrolling | | | —— |
| | | Comment display | | | To display comments of designated devices | 6.2.4 |
| INSERT (I) | Inserting in programs | | | | To insert new programs in existing programs | 6.4.1 |
| | Help | Insertion | Instruction help | Instruction display /selection | Display/selection of instructions that begin with designated characters | 6.2.3 |
| | | | | Step read | To designate step numbers and read programs | —— |
| | | | Move | | To move designated ranges in programs to designated positions | 6.4.2 |
| | | | Copy | | To copy designated ranges in programs to designated positions | 6.4.3 |
| | | Comment display | | | To display comments of designated devices | 6.2.4 |
| DELETE (D) | Deleting programs | | | | To delete programs of designated steps | 6.5.1 |
| | Help | Deletion | Range designation | | To delete designated ranges in programs | 6.5.2 |
| | | | NOP batch | | To batch-delete NOP instructions in programs up to the END instructions (NOPLF instructions are not deleted.) | 6.5.3 |
| | | Comment display | | | To display comments of designated devices | 6.2.4 |
| MONITOR (M) | List monitoring | | | | To read programs of designated steps and display the continuity of instructions, ON/OFF states of contacts, and current values | 6.6.1 |
| | Monitoring search | | | | To search and display OUT/SET/RST instructions using designated contact devices (Monitoring continues.) | 6.6.2 |
| | Device monitoring | | | | To display current values (including T/C set values) of the ON/OFF states of bit devices and word devices | 6.6.3 |
| | Help | Display change | | | To display values in designated formats or ASCII. | 6.6.4 |
| | | Monitor | List monitoring | | To read and monitor programs of designated steps | —— |
| | | | | | Same as the monitor search function. | —— |
| | | | Device monitoring | | To display current values (including T/C set values) of the ON/OFF state of bit devices and word devices | —— |
| | | | | | To set displays of offline switchs | 6.6.5 |
| | | Comment display | | | To display comments of designated devices | 6.2.4 |

(Continued on the next page.)

(Continued from the previous page)

| Modes (Mode Displays) | Functions | | | Purposes | Reference Sections |
|---|---|---|---|---|---|
| TEST (T) | Testing by list monitoring | | | To set(turn ON)/reset(turn OFF) bit devices<br>To change current values of word devices | 6.7.1<br>6.7.2 |
| | Test by device monitoring | | | To set/cancel offline switchs and separate designated bit devices utilized in OUT instructions from ACPU operation processing. Or to cancel separations | 6.7.5 |
| | | | | To set(turn ON)/reset(turn OFF) bit devices<br>To change current values of word devices | 6.7.3<br>6.7.4 |
| | Help | Display change | | To display values in designated formats or ASCII. | 6.6.4 |
| | | Test | List monitoring test | To read programs of designated steps, to set(turn ON)/reset(turn OFF) bit devices, and to change current values of word devices Same as the Test by list monitoring. | —— |
| | | | Device monitoring test | To display current values (including T/C set values) of the ON/OFF state of bit devices and word devices | —— |
| | | Comment display | | To display comments of designated devices | 6.2.4 |
| PARAMETER (P) | All clear of parameters | | | To only clear parameters in an ACPU | 6.8.1 |
| | Setting of parameters | | | To set memory capacities, timer counters, and latch ranges, etc of all kinds of parameters Or to change set values | 6.8.2 |
| | | | | To set entry codes<br>Or to change entry codes | 5.1.3<br>6.8.2 |
| OTHERS (O) | Change of T/C set values | | | To change set values of designated devices (T/C) | 6.9.1 |
| | PC check | Error step read | | To display the descriptions of errors that occur in an ACPU and the step numbers where errors occur | 6.9.2 |
| | | Program check | | To check duplex coil/instruction codes, etc. in programs | 6.9.3 |
| | PC system | Monitoring | Link monitoring | To display MELSECNET(II)/B link states | 6.9.4 |
| | | | Buffer memory batch monitoring | To monitor the buffer memory contents of designated addresses for the special-function modules of designated I/O numbers | 6.9.5 |
| | | | Clock monitoring | To monitor ACPU clocks (D9025 to D9027) | 6.9.6 |
| | | All clear | PC memory | To clear the entire memory of ACPUs and return to initial states | 6.9.7 |
| | | | Program | To clear currently selected programs (main/sub) | 6.9.8 |
| | | | Device memory | To clear all device memories except special D and special M and R | 6.9.9 |
| | | Switching | PC NO. setting | To switch applicable operating programs in all A8PUE modes. | 6.9.10 |
| | | | Main/sub switching | To switch applicable operating programs (main/sub) in all A8PUE modes | 6.9.11 |
| | | Others | Remote RUN/STOP | To forcibly switch ACPU execution states (RUN/STOP) | 6.9.12 |
| | | | Machine language reading /writing | To read and write in machine language for the memory of ACPUs | 6.9.13 |
| | PU setting | Program mode selection | | To set whether or not writing is executed when programs are in the RUN state, and whether or not only the MONITOR and TEST are utilized. | 6.9.14 |
| | | Continuity display | | To set whether or not the continuity of each instruction is displayed when the list monitoring function is used. | 6.9.15 |
| | | Buzzer setting | | To set whether or not buzzer is ON/OFF when a key is pressed. | 6.9.16 |

### 3.4.2 List of help functions

The following table shows the help functions for each mode. Operate the help function without using the reference section in accordance with the items in each help menu.

| Modes (Mode Displays) | Help Functions | Purposes | Help Operations | Normal Operations | Reference Sections |
|---|---|---|---|---|---|
| WRITE (W) | Instruction display/selection | • To input initials and display instruction names that correspond to such initials in lists<br>• To select instructions to be written from lists | O | — | 6.2.3 |
| | Step read | To designate step numbers in which instructions are written and to display programs | O | O | — |
| | Continuous writing (range) of NOP | To designate start and final step numbers and execute NOP batch writing | O | — | 6.2.1 |
| | Continuous writing (all) of NOP | To designate only start step numbers and execute NOP batch writing until the flag step | O | — | |
| | Comment display | To display comments of devices where the cursor is | O | — | 6.2.4 |
| READ (R) | Read by using step designation | To designate step numbers and display programs | O | O | — |
| | Read by using an instruction *1 | To input instruction names and search programs | O | O | — |
| | Read by using a device | To input device numbers and search programs | O | O | — |
| | Automatic scrolling | To scroll programs automatically in designated directions | O | O | — |
| | Comment display | To display comments of device where the cursor is | O | — | 6.2.4 |
| INSERT (I) | Instruction display/selection | • To input initials and display instruction names that correspond to such initials in lists<br>• To select instructions to be inserted from lists | O | — | 6.2.3 |
| | Read by using step designation | To designate step numbers in which instructions are inserted and to display programs | O | O | — |
| | Batch movement of programs | To designate step ranges and move in batch | O | — | 6.4.2 |
| | Batch copying of programs | To designate step ranges and copy them in batch | O | — | 6.4.3 |
| | Comment display | To display comments of devices where the cursor is | O | — | 6.2.4 |

*1: See (2) and (3) in the Explanations part of Section 6.3.2 for the instruction to be set.

(Continued on the next page.)

(Continued from the previous page)

| Modes (Mode Displays) | Help Functions | Purposes | Help Operations | Normal Operations | Reference Sections |
|---|---|---|---|---|---|
| DELETE (D) | Batch deleting of programs | To designate step ranges and delete them in batch | O | — | 6.5.2 |
| | NOP batch delete | To batch-delete NOP instructions | O | — | 6.5.3 |
| | Comment display | To display comments of devices where the cursor is | O | — | 6.2.4 |
| MONITOR (M) | Changing display formats | To change displayed numerical values to binary, octal, decimal, hexadecimal, and ASCII displays | O | — | 6.6.4 |
| | Read by using step designation | To designate step numbers that do list monitoring and display programs | O | O | — |
| | Monitoring search | To move the cursor to contact instructions and search the corresponding OUT, SET, and RST instructions | O | O | — |
| | Monitor device | To designate devices to be monitored and display devices and device states | O | O | — |
| | Setting of an offline switch display | To set whether or not there is an offline switch display during device monitoring | O | — | 6.6.5 |
| | Comment display | To display comments of devices where the cursor is | O | — | 6.2.4 |
| TEST (T) | Changing display formats | To change displayed numerical values to binary, octal, decimal, hexadecimal, and ASCII displays | O | — | 6.6.4 |
| | Read by using step designation | To designate step numbers in which list monitoring is tested and to display programs | O | O | — |
| | Device monitoring test | To designate devices which device monitoring is tested and display devices and device states | O | O | — |
| | Comment display | To display comments of devices where the cursor is | O | — | 6.2.4 |
| OTHERS (O) *2 | Changing display formats | To change displayed numerical values to binary, octal, decimal, hexadecimal, and ASCII displays | O | — | 6.6.4 |

*2: Can be executed only during buffer memory batch monitoring.

# 4. HANDLING AND NOMENCLATURE

This section tells how to handle the A8PUE and explains the part names

## 4.1 Precautions When Handling the A8PUE

The following precautions should be taken when handling the A8PUE:

(1) Since the case is made of plastic, do not drop the A8PUE or subject it to severe shocks.

(2) Do not disassemble the case (doing so could cause a malfunction).

(3) When not using the A8PUE, handle the RS-422 connector as follows:

1) Attach the cover to the RS-422 connector on the right side of the unit.

2) Attach the protective cap to the RS-422 connector on the rear side of the unit.

(4) When using the A8PUE, the RS-422 connector not used for connection to an ACPU should have its cover or protective cap on.

(5) Do not remove the connector cover of the extension interface on the upper side of the unit.

(6) Do not touch the RS-422 connector pins (doing so could cause a malfunction).

(7) Do not remove the cover of the A8PUE.
Do not loosen the screws that hold the cover (doing so could cause a malfunction).

(8) Always press the [CLEAR] key before disconnecting the A8PUE from an ACPU.

(9) Never use a cleaning agent such as thinner, alcohol, or freon.

(10) Press keys only with the fingers.
Using a sharp instrument, etc. could cause a malfunction.
Since the A8PUE beeps when a key is pressed, key operations can be confirmed by the sound.

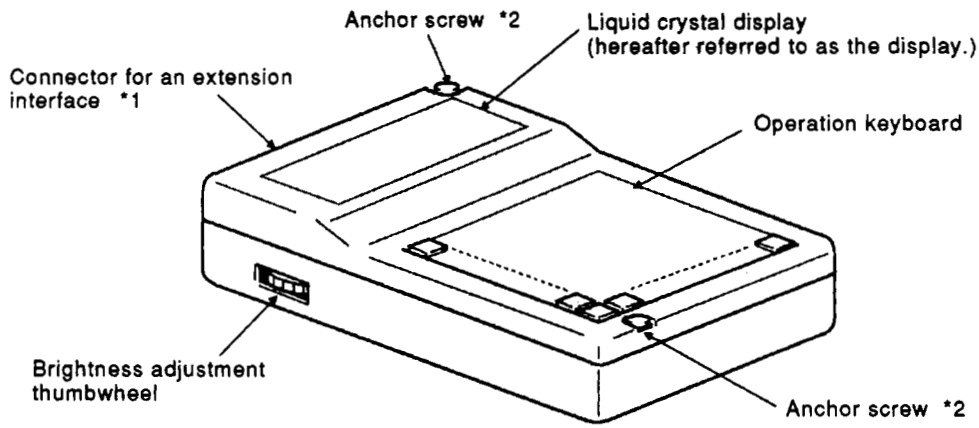
**IMPORTANT**

When designing the system, to protect the PC, make sure to provide a safety circuit outside that system.

## 4.2 Nomenclature

This section shows and explains the nomenclature of the A8PUE.

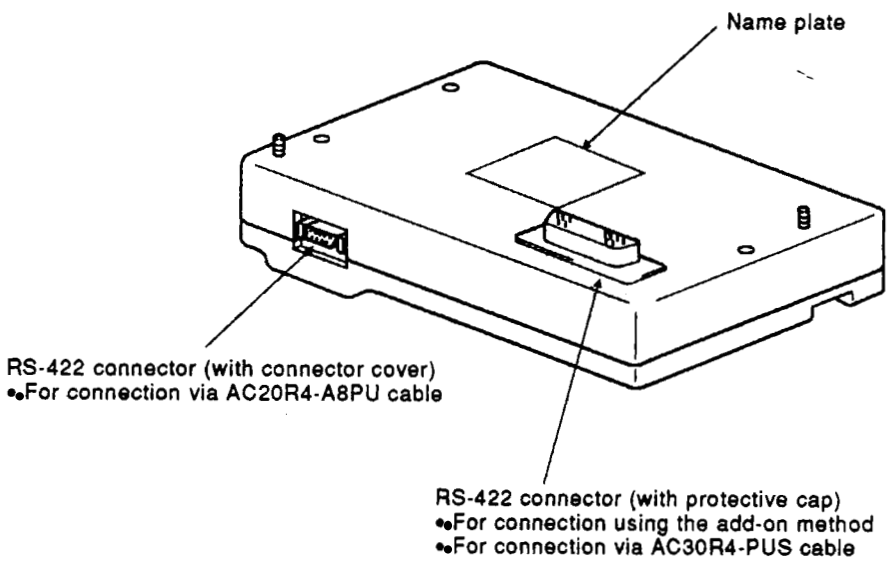
(Front view)




*1 The connector for an extension interface cannot be currently used. Do not remove the connector cover.

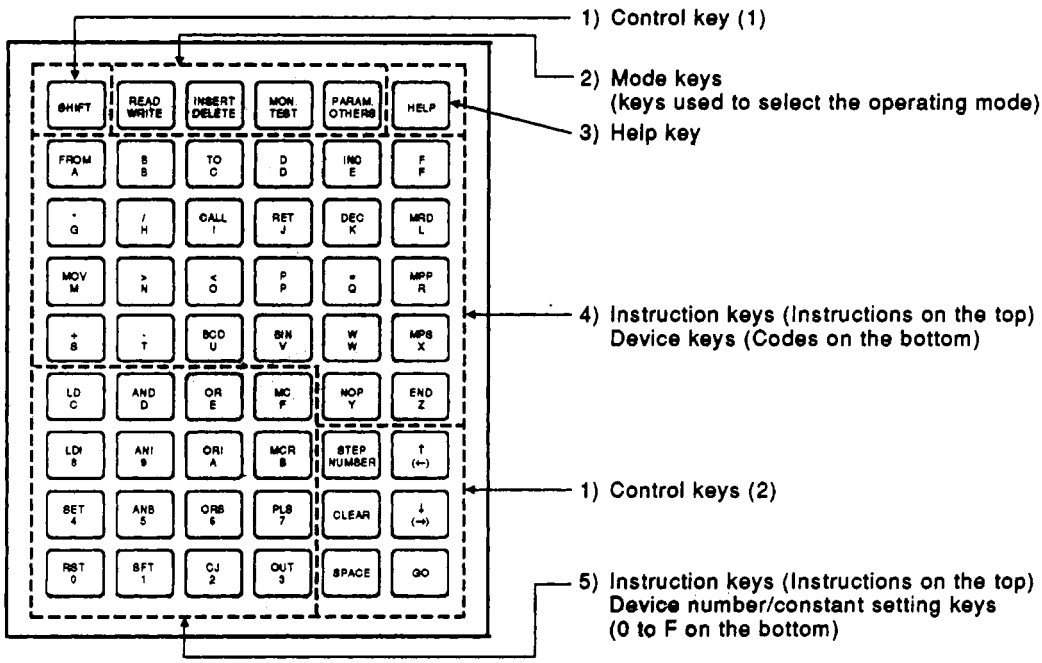
*2 The anchor screw (M3) tightening torque is (39 to 49 N·cm 4 to 5kg·cm).

(Rear view)


Name plate
RS-422 connector (with connector cover)
•For connection via AC20R4-A8PU cable
RS-422 connector (with protective cap)
•For connection using the add-on method
•For connection via AC30R4-PUS cable

## 4.3 Layout of the Keyboard and List of Key Functions

This section gives the layout of the A8PUE operating keyboard, and lists the keys and their functions.




| Numbers. | Names | Keys | Functions | Main Reference Sections |
|---|---|---|---|---|
| 1) | Control key | STEP NUMBER | Used to declare start inputting step numbers or for automatic scrolling. Sections 2 and 6 give examples of its use.<br>Used to make the bottom functions of a key valid.<br>The top/bottom setting can be confirmed by the display. | 2<br>6 |
| | | SHIFT | Used to make the top functions of a key valid.<br>The top/bottom setting can be confirmed by the display. | 5.4.1 |
| | | CLEAR | In modes other than PARAMETER, OTHERS, and help function modes:<br>Used to return to the mode selection state.<br>(All input instructions or device numbers will be cleared except the mode.)<br>Used to correct a mistake when the wrong key is input. | 6 |
| | | | In the PARAMETER mode:<br>Used to interrupt processing. After processing has been restarted, continue operations as before. | 6.8 |
| | | | Returns to the previous display in the OTHERS mode. | 6.9 |
| | | | When the help function is used:<br>Returns to the display when the [HELP] key was pressed. | 6 |
| | | | When disconnecting the A8PUE from an ACPU:<br>Processing of A8PUE during execution is interrupted.<br>Be sure to press the [CLEAR] key before disconnecting them. | 5.1.5 |
| | | SPACE | Used to input a blank between an instruction part and a device name | 6 |
| | | ↑ (←) ↓ (→) | Used to move the cursor (▶, ■) on the display or indicate the scrolling direction of a scrolling display. | 6 |
| | | GO | Pressed at the end of a series of key operations. Key operations are executed until this key is pressed. Press this key after confirming the contents of the series of key operation in the display. | 6 |

(Continues on the next page.)

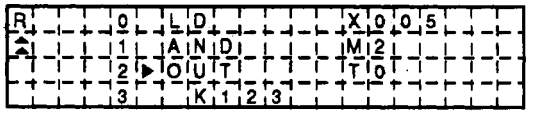
(Continued from the previous page)

| Numbers. | Names | Keys | Functions | Main Reference Sections |
|---|---|---|---|---|
| 2) | Mode keys | READ WRITE to PARAM. OTHERS | Used to select A8PUE modes. The top and bottom modes can be switched by pressing the [SHIFT] key. | 5.1.4 |
| 3) | Help key | HELP | Used to select the help function in the mode supporting a help function (Sections 2.5, 6.2 to 6.7, and 6.9.5. give details about applicable modes.) | 2.5<br>6 |
| 4) | Instruction/Device keys | FROM A to END Z | Used to input K/H when instructions, device names and constants are input. Pressing the [SHIFT] or [STEP NUMBER] keys enables switching the valid key area between the top and bottom. | 5.4.2<br>5.4.1 |
| 5) | Instruction/ Device number/ constant setting keys | RST 0 to MC F | Used to input instructions, device numbers, and constants. Pressing the [SHIFT] or [STEP NUMBER] keys enables switching the valid key area between the top and bottom. | |

- The following shows the movements between the steps, between menu items, and in the input area.

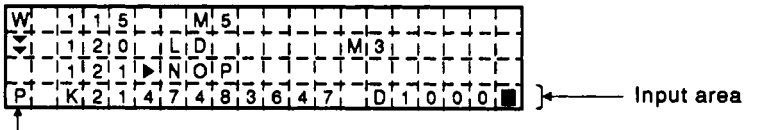
(1) Repeated pressing of the cursor keys move the cursor in the designated key direction.

(2) Movement between menu items or between steps
To display the programs immediately before or after a currently displayed instruction, press the [ ↑ ]/[ ↓ ] keys.
[ ↑ ] : Program immediately before (the program next to step 0 will not change)
[ ↓ ] : Program immediately after (the program next to the final step will not change)




"▶" can move in a designated direction (up and down) by pressing [ ↑ ] or [ ↓ ] key.

(3) Movement in the input area
Press the [(←)]/[(→)] keys to move the cursor among instruction names, sources, and destinations.




"■" can move in a designated direction (right and left) by pressing the [(←)] or [(→)] key.

In this manual, A8PUE key operations are abbreviated as follows:

(1) [Key 1] → [Key 2] → ···· [Key n]: This means all keys between [Key 1] and [Key n] are pressed in order.

(2) [Key 1] + [Key 2]: This means that both [Key 1] and [Key 2] are pressed simultaneously.

(3) Keys such as [RST / 0] and [MOV / M] keys which are used for dual purposes (instruction and device/constant input), and the control keys of the A8PUE are referred to as follows:

(a) The mode instruction or alphanumeric character which corresponds to the purpose of those keys.

Example:
[RST / 0] → [RST] or [0], [MOV / M] → [MOV] or [M]

(b) When an instruction input is explained, the explanation uses only the instruction symbol. (The alphanumeric character is omitted.) When an alphanumeric character input is explained, the explanation uses only the alphanumeric character. (The instruction name is omitted.)

Example:
When the [MOV / M] key is referred to:
When an instruction input is explained:[MOV]
When an alphanumeric character input is explained: [M]

## 4.4 Maintenance

Except for the brightness control, the A8PUE has no components which require inspection or replacement.

Rules for storing the A8PUE:

(1) Do not store the A8PUE in the following environments:
- Where ambient temperature is outside the range of −20 °C to 70 °C.
- Where ambient humidity is outside the range of 10 to 90 % RH.
- Where condensation occurs due to sudden temperature changes.
- Anywhere the A8PUE might be exposed to wind, rain, or the direct sunlight.
- Anywhere with excessive amounts of conductive powders (such as dust, dirt, and iron filings) or corrosive gases, oil mist, and salt.

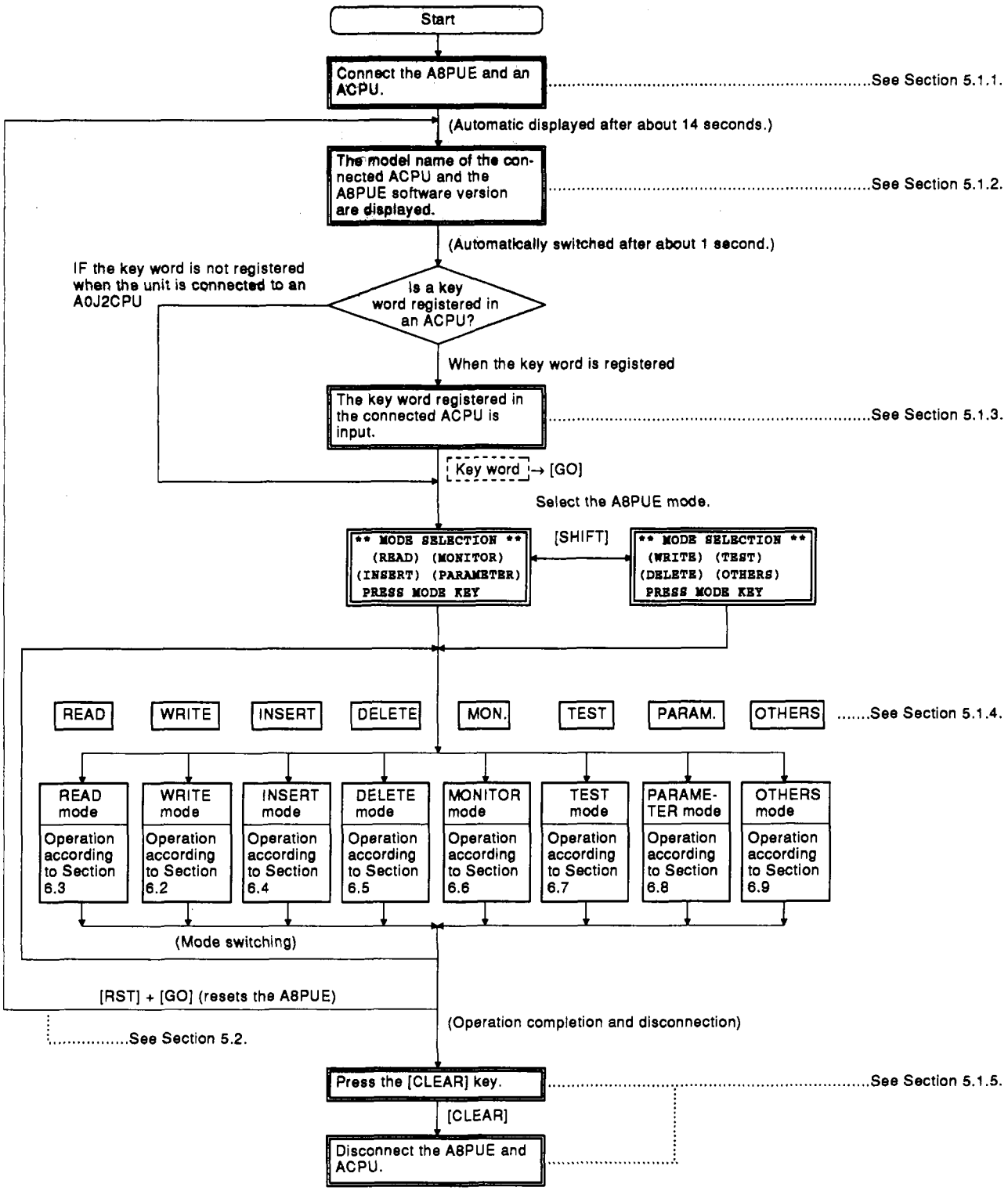
(2) When storing, make sure the protective cap is on the RS-422 connector on the rear side of the A8PUE.

(3) When storing, make sure the cover is on the RS-422 connector on the right side of the A8PUE.

# 5. OPERATING PROCEDURES

This section explains the operating procedures from the time the A8PUE is connected to an ACPU until it is disconnected, as well as related window displays and basic operations.

## 5.1 Procedure From the Beginning to the End of the Operation

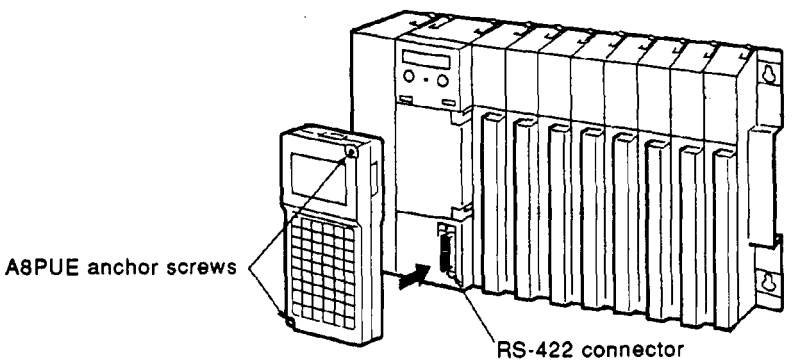


### 5.1.1 Connecting the A8PUE to an ACPU

Although the A8PUE can be connected to an ACPU even while it is running, Mitsubishi recommends connecting the A8PUE to an ACPU while it is in the STOP state. If it must be connected while in the RUN state, make sure to insert the connector properly.

(1) Connection using the add-on method
In this method, the A8PUE is installed directly onto the ACPU module.

However, the add-on method cannot be used for connecting an A1S and A2CCPU(C24/PRF) because screws cannot be tightened.
Connect then using the hand-held method.




The connection procedure is as follows:

1) Remove the cover of the ACPU's RS-422 connector.
2) Remove the RS-422 connector protection cap from the rear of the A8PUE.
Put the protection cap in a safe place.
3) Connect the A8PUE to the RS-422 connector of an ACPU, and firmly tighten the A8PUE anchor screws.
Tighten the anchor screws at a torque of 39 to 49 N·cm (4 to 5 kg·cm).
4) About 14 seconds after the connection is completed, both the A8PUE software version and the ACPU model name will be displayed.

(2) Connection using the hand-held method
Connect the A8PUE to an ACPU by using the RS-422 cable.

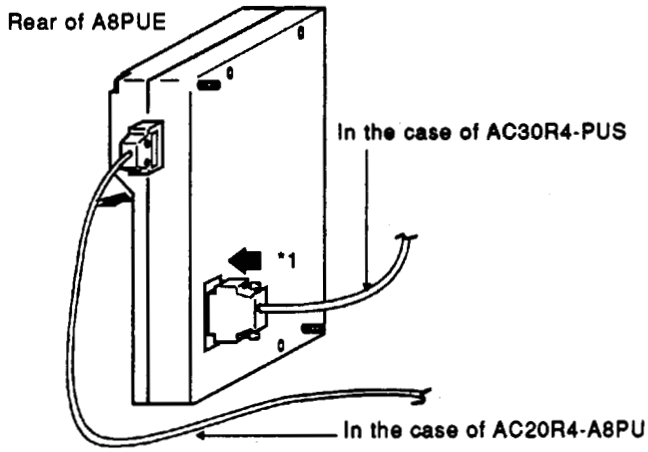
The connection procedure is as follows:

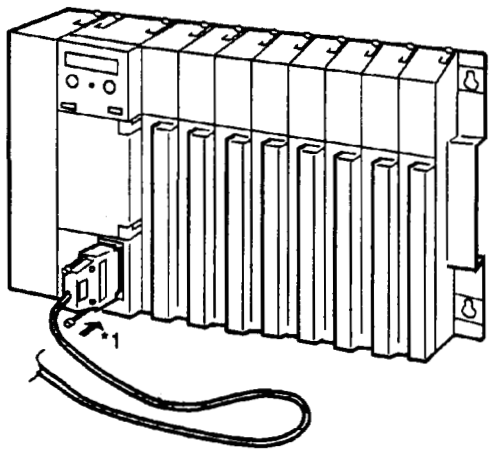
1) Remove the cover of the ACPU's RS-422 connector.

2)-1 When using an AC30R4-PUS cable
Remove the RS-422 connector protection cap from the rear of the A8PUE.
Put the protection cap in a safe place.

2)-2 When using an AC20R4-A8PU cable
Open the cover of the RS-422 cable on the right side of the A8PUE.

3) Connect the A8PUE to the ACPU using the RS-422 cable.



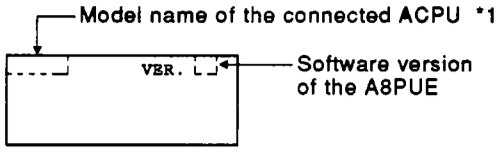



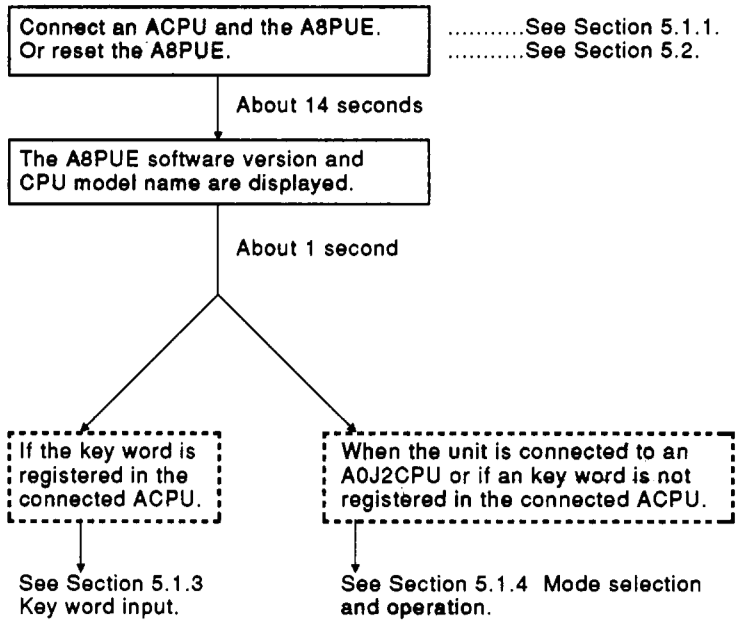

*1 Firmly tighten a anchor screws of the connector.

4) About 14 seconds after the connection is completed, both the A8PUE software version and the model name of the connected ACPU will be displayed.

### 5.1.2 A8PUE software version and CPU model name display

After (a) completing the connection with an ACPU, or (b) resetting (pressing the [RST] + [GO] keys) the A8PUE, the A8PUE software version and CPU model name are shown in the A8PUE display.
Confirm the display contents as shown below.



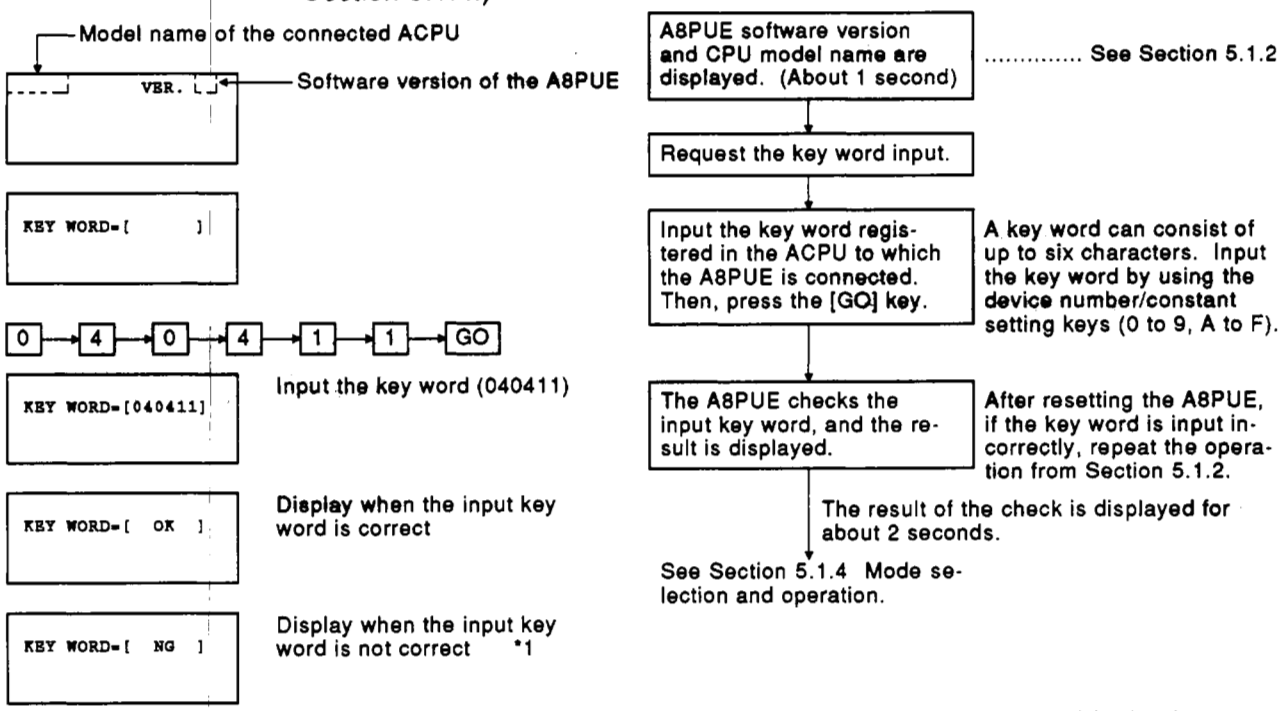


Connect an ACPU and the A8PUE.
Or reset the A8PUE.
.....See Section 5.1.1.
.....See Section 5.2.
About 14 seconds
The A8PUE software version and CPU model name are displayed.
About 1 second
If the key word is registered in the connected ACPU.
When the unit is connected to an A0J2CPU or if an key word is not registered in the connected ACPU.
See Section 5.1.3 Key word input.
See Section 5.1.4 Mode selection and operation.

*1: The following list shows the connectable ACPUs and model names which can be displayed.

| Connected ACPUs | Model Names |
|---|---|
| A0J2CPU | A0J2 |
| A0J2HCPU | A2 |
| A1CPU | A1 |
| A1NCPU | A1 |
| A1SCPU(-S1) | A1S |
| A2CPU(-S1) | A2 |
| A2NCPU(-S1) | A2 |
| A2CCPU(C24/PRF) | A2 |
| A2ACPU(-S1) | A2A |
| A3CPU | A3 |
| A3NCPU | A3 |
| A3ACPU | A3A |
| A3HCPU | A3H |
| A3MCPU | A3M |
| A3VCPU | A3 |

### 5.1.3 Key word input

When a connected ACPU or the PC NO. of the ACPU to be operated is switched, if the key word is registered in the ACPU, the A8PUE requests an input of the registered key word.
Input the key word registered in the ACPU. Then, press the [GO] key.
If the key word is not registered in the ACPU or the A8PUE is connected to an A0J2CPU, the operation shown in this section is unnecessary. (The A8PUE automatically goes into the mode and function selection shown in Section 5.1.4.)


Model name of the connected ACPU
VER.
Software version of the A8PUE
KEY WORD=[ ]
0 4 0 4 1 1 GO
KEY WORD=[040411]
Input the key word (040411)
KEY WORD=[ OK ]
Display when the input key word is correct
KEY WORD=[ NG ]
Display when the input key word is not correct *1
A8PUE software version and CPU model name are displayed. (About 1 second)
See Section 5.1.2
Request the key word input.
Input the key word registered in the ACPU to which the A8PUE is connected. Then, press the [GO] key.
A key word can consist of up to six characters. Input the key word by using the device number/constant setting keys (0 to 9, A to F).
The A8PUE checks the input key word, and the result is displayed.
After resetting the A8PUE, if the key word is input incorrectly, repeat the operation from Section 5.1.2.
The result of the check is displayed for about 2 seconds.
See Section 5.1.4 Mode selection and operation.

*1: If the key word which was input does not correspond with the key word registered in the ACPU, only the following operations can be done (Section 6 gives details):

- MONITOR mode
  - Device monitoring ..... See Section 6.6.3
  - Monitoring the offline switch ..... See Section 6.6.5
- TEST mode
  - Test by device monitoring ..... See Sections 6.7.3 and 6.7.4
  - Setting/canceling the offline switch ..... See Section 6.7.5
- OTHERS mode
  - Error step read ..... See Section 6.9.2
  - Monitoring/switching the PC system ..... See Sections 6.9.4, 6.9.5, 6.9.6, 6.9.10, and 6.9.11
  - PU setting ..... See Sections 6.9.14, 6.9.15, and 6.9.16

**POINTS**

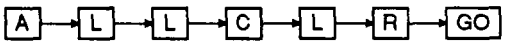
- When operating in modes other than MONITOR, TEST, and OTHERS, if the key word registered in the ACPU is unknown, see the "PC memory all clear" operation shown on the next page. And then, clear (delete) the unknown registered key word. If "PC memory all clear" is executed, all other user data (such as sequence programs) will also be cleared.
- Section 6.8.2 gives details about when (a) the key word registered in an ACPU is changed, or (b) a new key word is registered.

All clear of parameters and a sequence programs as well as key words registered in an ACPU is enabled by the following operation when a key word input is requested:

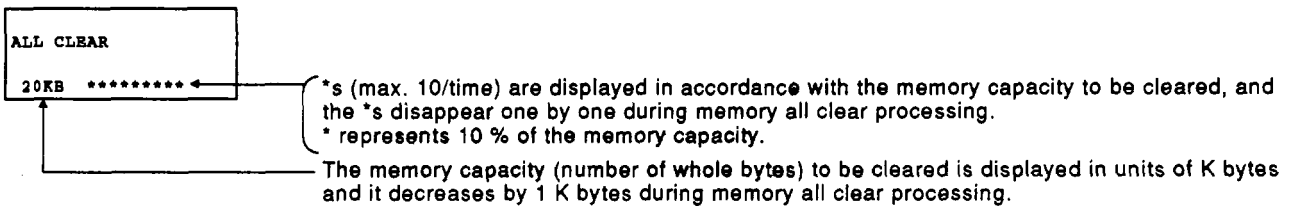
(Procedure 1) Key word input request is displayed.

(Procedure 2) ACPU STOP operation
Put the ACPU in the STOP state.

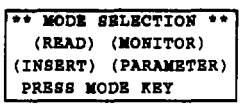
(Procedure 3) PC memory all clear operation
Input "ALLCLR", and then press the [GO] key.




(Procedure 4) PC memory all clear processing display
When memory all clear processing by the A8PUE begins, the screen shown on the left is displayed, and the display on the "*" line and the number of whole bytes changes sequentially.


ALL CLEAR
20KB *********
*s (max. 10/time) are displayed in accordance with the memory capacity to be cleared, and the *s disappear one by one during memory all clear processing.
* represents 10 % of the memory capacity.
The memory capacity (number of whole bytes) to be cleared is displayed in units of K bytes and it decreases by 1 K bytes during memory all clear processing.

(Procedure 5) Completion of PC memory all clear processing
When memory all clear processing by the A8PUE has been completed, a buzzer sounds.
The screen is switched to the one shown on the left. (State before selecting the A8PUE mode)




(Procedure 6) The next operation begins
Select the mode in accordance with Section 5.1.4, and start the next operation.

### 5.1.4 Mode selection and operation

After displaying the CPU model name and A8PUE software version and executing the key word input operation, perform the A8PUE mode selection to prepare for the operations shown Section 6.

(1) Using the mode keys to select and change modes
To select the mode, press the corresponding mode key. Then, operations can be performed in the selected mode as shown in Section 6.
When the operations shown in Sections 6.2 to 6.9 are executed, the mode can be changed. Therefore, the operation that is executed can be interrupted, and the mode can be changed. *1
The relationship between mode keys, modes and their corresponding sections in this manual is shown below:

| | | |
|---|---|---|
| [READ] | READ mode | Section 6.3 |
| [WRITE] | WRITE mode | Section 6.2 |
| [INSERT] | INSERT mode | Section 6.4 |
| [DELETE] | DELETE mode | Section 6.5 |
| [MON.] | MONITOR mode | Section 6.6 |
| [TEST] | TEST mode | Section 6.7 |
| [PARAM.] | PARAMETER mode | Section 6.8 |
| [OTHERS] | OTHERS mode | Section 6.9 |

*1: Pressing the mode key is always valid.
Input data (except step numbers) is cleared by pressing the mode key.
The input mode returns to the initial state.

(2) Selecting the write enabled/disabled function when the ACPU is in the RUN state
Write enabled/disabled when in the ACPU is in the RUN state is decided by the program mode selection (see Section 6.9.14) in the OTHERS mode.

(3) To operate in all modes
The operations shown in Section 6 are enabled by the mode selected in (1) and the function selected in (2).
The mode, the ACPU operating format (RAM/EEPROM/ROM), and the ACPU's operation enabled/disabled state are shown on the top of the first page of each operation explanation.
Because the operating methods in Section 6 are explained by using names which correspond with the purposes of the various modes, perform operations in accordance with these explanations.

See Section 7 when an error message is displayed during operations.

### 5.1.5 Disconnecting the A8PUE from an ACPU

(1) Disconnection when the add-on method is used
The disconnection procedure is as follows:

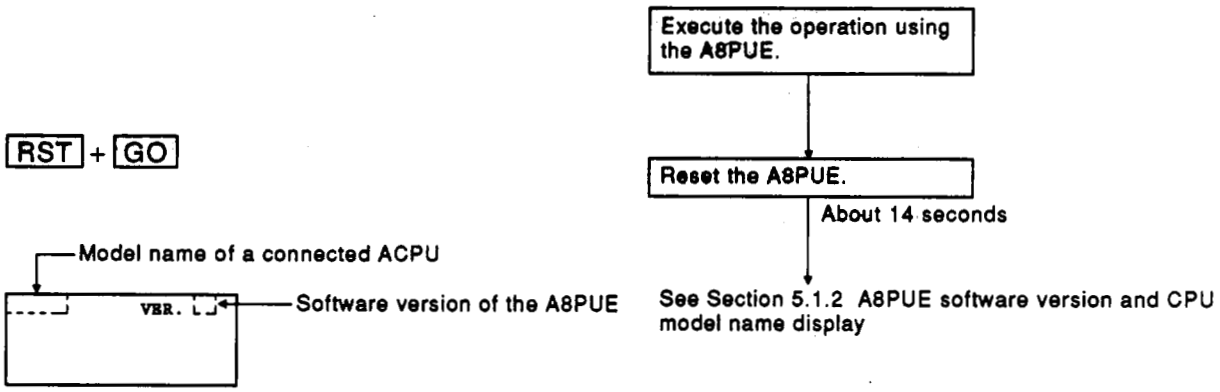
1) Press the [CLEAR] key.
2) To disconnect the A8PUE from an ACPU, unscrew the A8PUE anchor screws.
3) Disconnect the A8PUE from the ACPU.
4) Put the stored protective cap back on the RS-422 connector on the rear side of the A8PUE.
5) Put the cover on the RS-422 connector of the ACPU.
6) Store the A8PUE carefully.

(2) Disconnection when the hand-held method is used
The disconnection procedure is as follows:

1) Press the [CLEAR] key.
2) Unscrew the connector anchor screws of the ACPU, and remove the RS-422 cable from the ACPU.
3) Put the cover on the RS-422 connector of the ACPU.
4) Method for removing RS-422 cable from the A8PUE
   (a) When AC30R4-PUS cable is used
   Unscrew the connector anchor screw of the A8PUE, and remove the RS-422 cable from the A8PUE.
   Put the stored protective cap on the RS-422 connector on the rear side of the A8PUE.
   (b) When AC20R4-A8PU cable is used
   Remove the RS-422 cable from the A8PUE.
   Put the stored protective cap on the RS-422 connector in the right side of the A8PUE.
5) Store the A8PUE and RS-422 cable carefully.

## 5.2 How to Reset the A8PUE

This section tells how to reset the A8PUE during operations.
After resetting the A8PUE, resume operations in accordance with Section 5.1.2 A8PUE software version and CPU model name display.

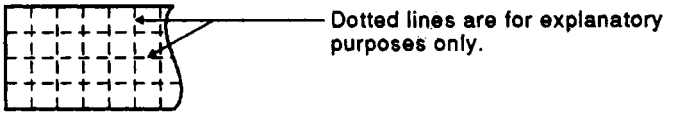

Execute the operation using the A8PUE.
RST + GO
Reset the A8PUE.
About 14 seconds
Model name of a connected ACPU
VER.
Software version of the A8PUE
See Section 5.1.2 A8PUE software version and CPU model name display

## 5.3 Display Adjustments and Display Format

This section explains how to adjust the display and the display format of the A8PUE.

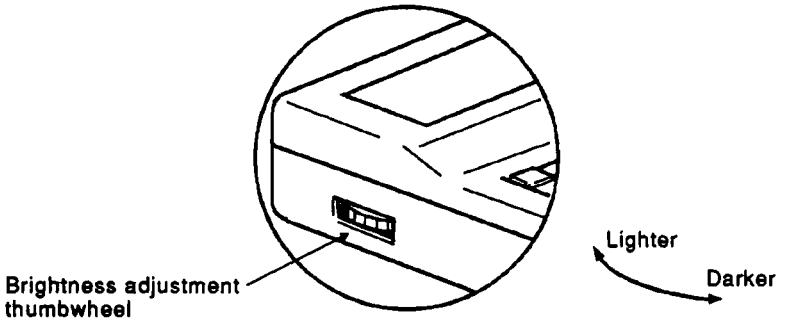
**REMARK**

The position where each character is shown on the display is indicated by the dotted lines in the display explanations given below.




### 5.3.1 How to adjust the brightness

After starting up the A8PUE, the brightness of the display can be adjusted by turning the brightness adjustment thumbwheel on the left side of the unit. Adjust as necessary.




### 5.3.2 Display backlight ON/OFF states

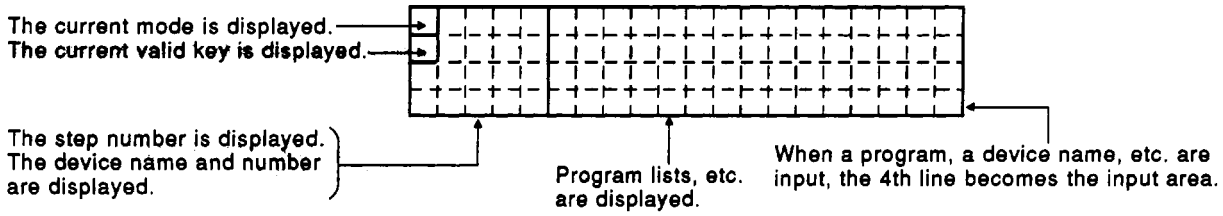
The backlight is ON/OFF in the following cases:
When the backlight is ON, perform operations as necessary.

(1) Goes ON
- When the A8PUE is started up
- When starting/restarting key inputting

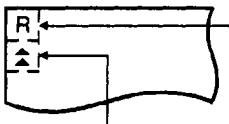
(2) Goes OFF
- If a key is not input for 10 minutes or more

### 5.3.3 Display format

This section explains the positions and contents of all data displayed on the A8PUE.
Depending on the mode, the actual display may sometimes differ from the following explanation. Section 6 gives details.


The current mode is displayed.
The current valid key is displayed.
The step number is displayed.
The device name and number are displayed.
Program lists, etc. are displayed.
When a program, a device name, etc. are input, the 4th line becomes the input area.

(1) Mode and valid key displays
The following shows the mode display after selecting the A8PUE mode and the display of the valid key that shows the validity of top/bottom of all dual-purpose keys.




The currently selected mode is displayed.

(Mode display) (Mode)
- R..... READ mode
- W..... WRITE mode
- I..... INSERT mode
- D..... DELETE mode
- M..... MONITOR mode
- T..... TEST mode
- P..... PARAMETER mode
- O..... OTHERS mode

The current valid key is displayed. (Sometimes not displayed.)

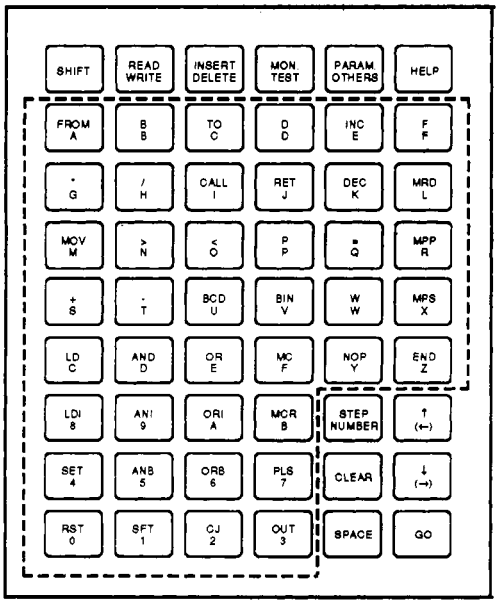
The valid keys are within the dotted frame in the figure on the left figure.

( ▲▲ : Top is valid.
▼▼ : Bottom is valid. )

(Example) In the case of the [FROM A] key

Top in the dotted line is valid..... [FROM]

Bottom in the dotted line is valid.... [A]

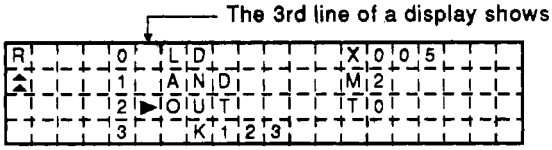



**POINT**

A valid key (top/bottom) can be switched by using the [SHIFT] or [STEP NUMBER] keys.
See Section 5.4.1.

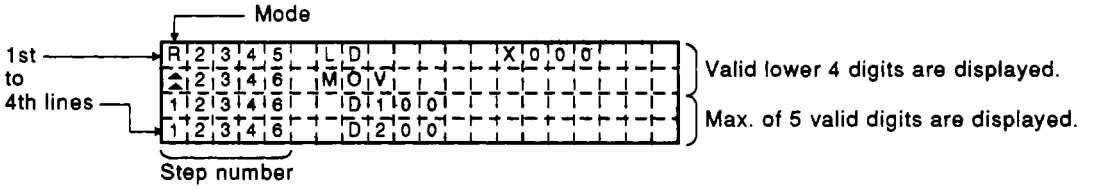
(2) Cursor display
The A8PUE controls the cursor display when data is input and " ■ " is shown at the cursor position. See (6) below.
However, when the cursor is moved onto a display character, the display character and " ■ " are displayed alternately.

(3) Operating line display
When a program list is displayed, the operating line is indicated by " ▶ " immediately after the step number.




Use the [ ↑ ] or [ ↓ ] keys to move " ▶ " in a designated direction (up or down).

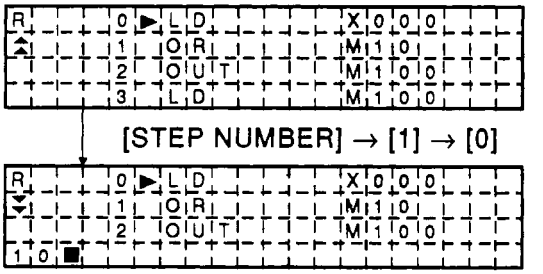
(4) Step number display
The step number is displayed in decimal.
1st to 2nd lines of the display show the valid lower 4 digits.
3rd to 4th lines of the display shows the max. 5 valid digits of the step number.



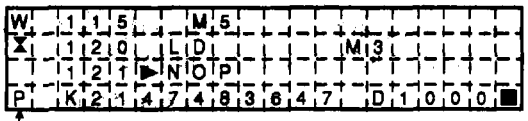

(5) Device display
If a basic instruction and an application instruction have 2 or more devices, they are displayed by the same step number.
See (4) above.

(6) Keyed-in data display
Keyed-in data is displayed at the cursor position sequentially and the cursor moves to right sequentially.
" ■ " is displayed at the cursor position.

(Example) When [STEP NUMBER] → [1] → [0] are input



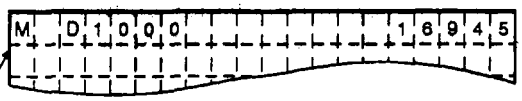
(7) Left shift display when a program is input
When a program is input, before the [GO] key is pressed, the program is displayed on the 4th line (at the bottom) of the display.
If a program cannot be displayed on 1 line, whenever a key is pressed, the contents of the displayed 4th line are shifted sequentially to the left.
(The input contents are stored internally.)



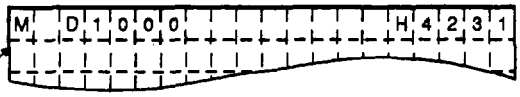

This is a display example when DMOVP K2147483647 D1000 has been input.

(8) Numerical value display
The numerical value display of each device and the constant display when a program is input are as follows.
Both are examples: for details, see Section 6.

1) Decimal display
Only valid digits are displayed by zero suppress.

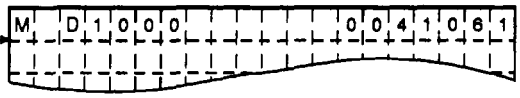



2) Hexadecimal display
4 digits are displayed without zero suppress.

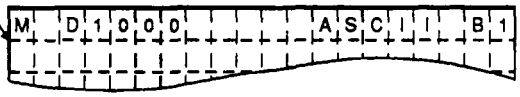



3) Octal display
6 digits are displayed without zero suppress.

(Example)
D1000 [4231H]




4) ASCII format display
The numerical value of a designated device is controlled by 1 byte, and the numerical value is considered as a character code. And then, a corresponding character is displayed.
Then, if the numerical value is other than 20H to 7FH, A0H to DFH, dot ".." is displayed.




(9) Error message display
Error messages are displayed on the 4th line of the display.
When an error message is displayed, take corrective actions in accordance with Section 7.
The displayed error message can be cleared by pressing a key. The unit returns to the state before the error message was displayed.

## 5.4 Basic Key Operations

This section explains the basic key operations after starting up the A8PUE.

### 5.4.1 Valid key switching (top/bottom)

When starting A8PUE operations, the validity of the top/bottom of dual-purpose keys is displayed on the left side of the second line of the display (see Section 5.3.3).
Valid keys are controlled and displayed by the A8PUE. However, users can switch valid key functions by pressing the following keys:

[SHIFT] * : Top key is valid.
[STEP NUMBER] : Bottom key is valid.

* Even when the bottom part of a dual-purpose key is valid, the following keys can be pressed. (The [SHIFT] key does not have to be pressed.)
  - Comparison symbol key when a comparison operation instruction is input .....[ > ], [ < ], [ = ]
  - Minus key of a source data area of an instruction .....[ − ]

After setting the different modes, valid keys are switched as follows:

| | |
|---|---|
| The READ mode, the WRITE mode, and the INSERT mode | : top part of the key is valid (When the cursor position in the set value or device step in the WRITE mode, the bottom part of the key is valid.) |
| MONITOR mode and TEST mode | : bottom part of the key is valid |
| PARAMETER mode and OTHERS mode | : the bottom part of the key is always valid |
| Help function in each mode | : the bottom part of the key is always valid (When INSTRUCTION READ is selected in the help of the READ mode, the top part of the key is valid.) |

If the [SHIFT] key is pressed and a valid key is switched, the key switched before one of the following operations becomes valid.

- The mode key is input. .....
- A control key is input. ..... } See Section 4.3.
- A8PUE is reset. ..... See Section 5.2.

### 5.4.2 How to Input Instructions

The methods of inputting instructions to the A8PUE are as follows:

1) To use the instruction shown on a key, press the instruction key.
2) Press the alphanumeric character keys that correspond to each character of an instruction sequentially.
3) Select and input an instruction by using the help function.

How to input instructions using 1) and 2) is shown below.
Section 6.2.3 explains how to input instructions using 3).

**POINT**

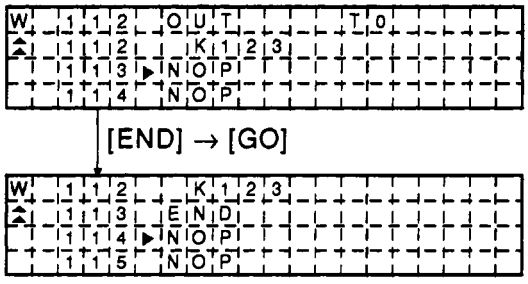
When inputting an instruction, the input contents are displayed on the 4th line (at the bottom) of the display.
When the [SPACE] key has to be pressed in the following explanation, if a blank is inserted automatically between an input instruction and the cursor position, the [SPACE] key does not have to be pressed.
See the examples for each explanation.

(1) Only in the case of an instruction code

1) When inputting a keyboard instruction

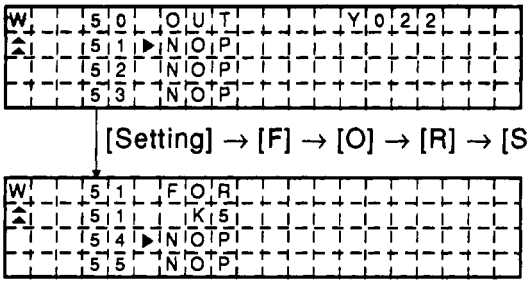
[Instruction key] → [GO]

(Example) When [END] is input




2) When inputting a non-keyboard instruction

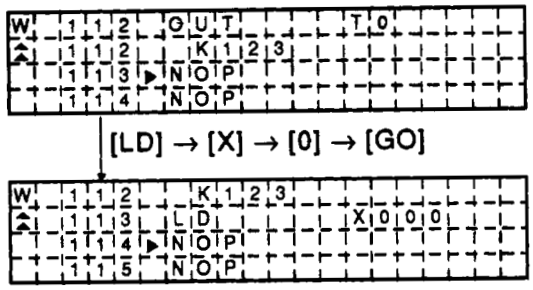
(Example) When FOR K5 is input



(2) In the case of an instruction code and a device

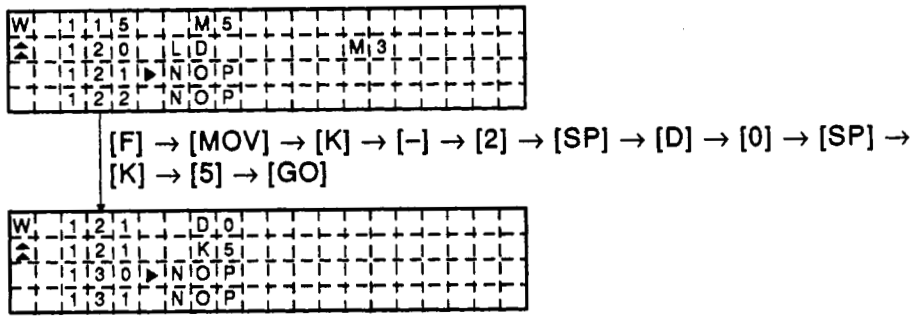
Instruction → [SP] → Device → Device number → [GO]

(Example) When LD X0 is input

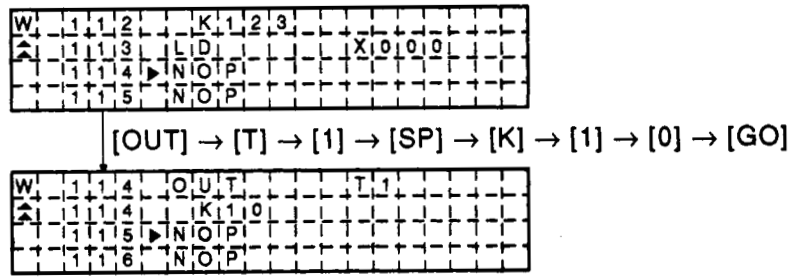



(3) In the case of an instruction other than the above
Press the [SPACE] key between an instruction, a device, source data, and destination data.

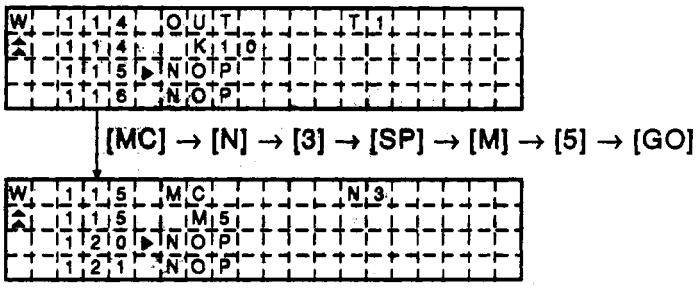
(Example 1) When FMOV K-2 D0 K5 is input



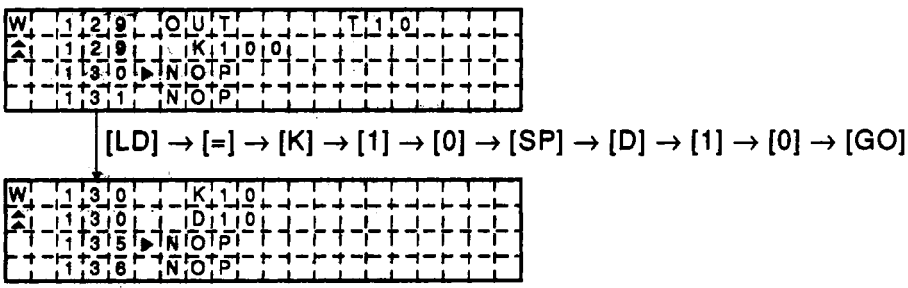

(Example 2) When OUT T1 K10 is input



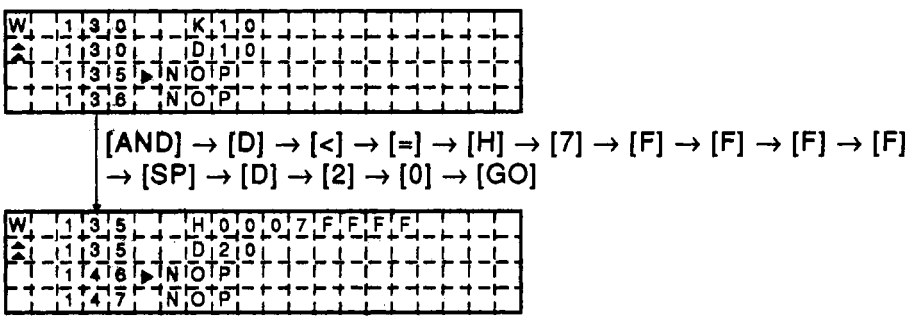
(Example 3) When MC N3 M5 is input




(Example 4) When LD=K10D10 is input




(Example 5) When ANDD<=H7FFFF D20 is input



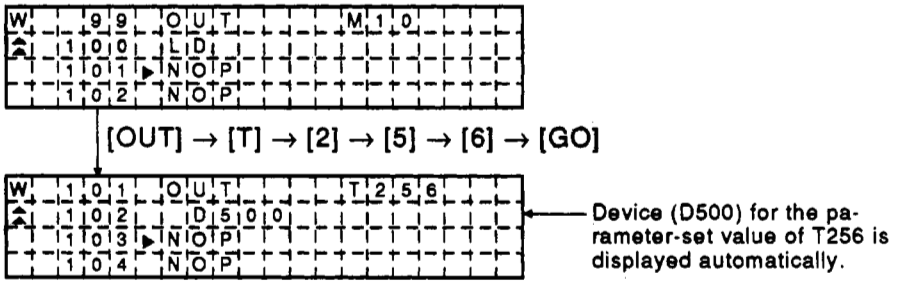

(4) Handling devices M, L, and S

For the devices M, L, and S in the TEST, MONITOR, WRITE, and INSERT modes, the display varies in accordance with the contents set by parameters.

For example, if LD L0 is input when the parameter settings are M0 to 999 and L1000 to L2047, LD M0 will be displayed.

(5) In the case of the extension timer/extension counter of an A2A and A3ACPU
If an extension timer (T256 to T2047) or extension counter (C256 to C1023) is input as a first device of an instruction, input the first device and a device number.

(Example) When OUT T256 D500 is input




**POINT**

When an extension timer or extension counter is used, be sure to set 257 points or more and device (D, W, R) for the set value at a parameter, as well as the timer or counter (see Section 6.8.2).

### 5.4.3 Corrective action when a wrong key has been pressed

(1) Before pressing the [GO] key, press the [CLEAR] key, and redo the operation.
Pressing the [CLEAR] key clears the previously-input instruction and its device number, and returns the A8PUE WRITE mode to its previous state. Then, the operation can be redone.

(2) When the [GO] key is pressed, redo the previous operation.

**REMARK**

If the [CLEAR] key is pressed while the A8PUE is in the PARAMETER mode, processing stops. When the operation is continued, continue the pressing the keys as before.

## 5.5 Functions of the A8PUE on a MELSECNET(/B) Data Link System

The A8PUE can execute testing, reading, writing, and monitoring of programs for the PC CPU of another station connected to the MELSECNET(/B) data link system (hereafter called MELSECNET).

(1) Stations that can access another station on MELSECNET are in the following ranges:

(a) When the A8PUE is connected to a master station
The self and all local stations can be accessed.
Remote I/O stations cannot be accessed.
Set and perform monitoring and testing for the I/O module of a remote I/O station at the self (master station).

(b) When the A8PUE is connected to a local station
The self and master stations can be accessed. Other local stations and remote I/O stations cannot be accessed.

(c) When the A8PUE is connected to a remote I/O station
The self and master stations can be accessed. Other local stations and remote other I/O stations cannot be accessed.

(2) Functions when accessing another station on MELSECNET

(a) When connecting the A8PUE to a master station/local station and accessing the self
It is possible to operate all functions.

(b) When connecting A8PUE to a remote I/O station and accessing the self
Link monitoring (see Section 6.9.4) and PC number setting (see Section 6.9.10) are possible.

(c) When connecting the A8PUE to a master station and accessing a local station
It is possible to operate functions that are the same as when the A8PUE was connected to the self - except for link monitoring (see Section 6.9.4) and PC memory all clear (see Section 6.9.7).

(d) When connecting the A8PUE to a local station and accessing a master station
It is possible to operate functions that are the same as when the A8PUE was connected to the self - except for link monitoring (see Section 6.9.4) and PC memory all clear (see Section 6.9.7).

(e) When connecting the A8PUE to a remote I/O station and accessing a master station
It is possible to operate functions that are the same as when the A8PUE was connected to the self - except for link monitoring (see Section 6.9.4) and PC memory all clear (see Section 6.9.7).

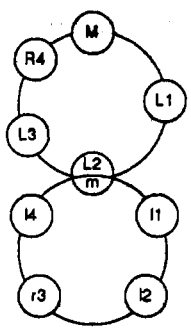
(3) The PC number setting when accessing another station on MELSECNET

(a) When connecting the A8PUE to a remote master station/local station/I/O station and accessing the self
Set "FF" to PC number.

(b) When connecting the A8PUE to a master station and accessing a local station
Set the station number of the local station to be accessed to PC number. For example, when accessing the third local station, set the PC number to 3.

(c) When connecting the A8PUE to a remote local station/I/O station and accessing a master station
Set the PC number to 0.

Fig. 5.1 shows the ranges in which the A8PUE can access other stations and the PC numbers when a 3-tier system has been established on MELSECNET.




M : Master station (tier 1)
L1 : Local station (tier 2)
L2/m : Local station, Master station (tier 2)
L3 : Local station (tier 2)
R4 : Remote I/O station (tier 2)
I1 : Local station (tier 3)
I2 : Local station (tier 3)
r3 : Remote I/O station (tier 3)
I4 : Local station (tier 3)

FF, 0 to 64 : Can be accessed (The numerals show PC numbers to be set.)
× : Cannot be accessed

| Station to which the A8PUE is connected | Accessable other stations (PC numbers) | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Self | M | L1 | L2/m | L3 | R4 | I1 | I2 | r3 | I4 |
| M | FF | 0 | 1 | 2 | 3 | × | × | × | × | × |
| L1 | FF | 0 | — | × | × | × | × | × | × | × |
| L2/m | FF | 0 | × | — | × | × | 1 | 2 | × | × |
| L3 | FF | 0 | × | × | — | × | × | × | × | × |
| R4 | FF | 0 | × | × | × | — | × | × | × | × |
| I1 | FF | × | × | 0 | × | × | — | × | × | × |
| I2 | FF | × | × | 0 | × | × | × | — | × | × |
| r3 | FF | × | × | 0 | × | × | × | × | — | × |
| I4 | FF | × | × | 0 | × | × | × | × | × | — |

**Fig. 5.1 Ranges in which the A8PUE can access other stations**

**REMARK**

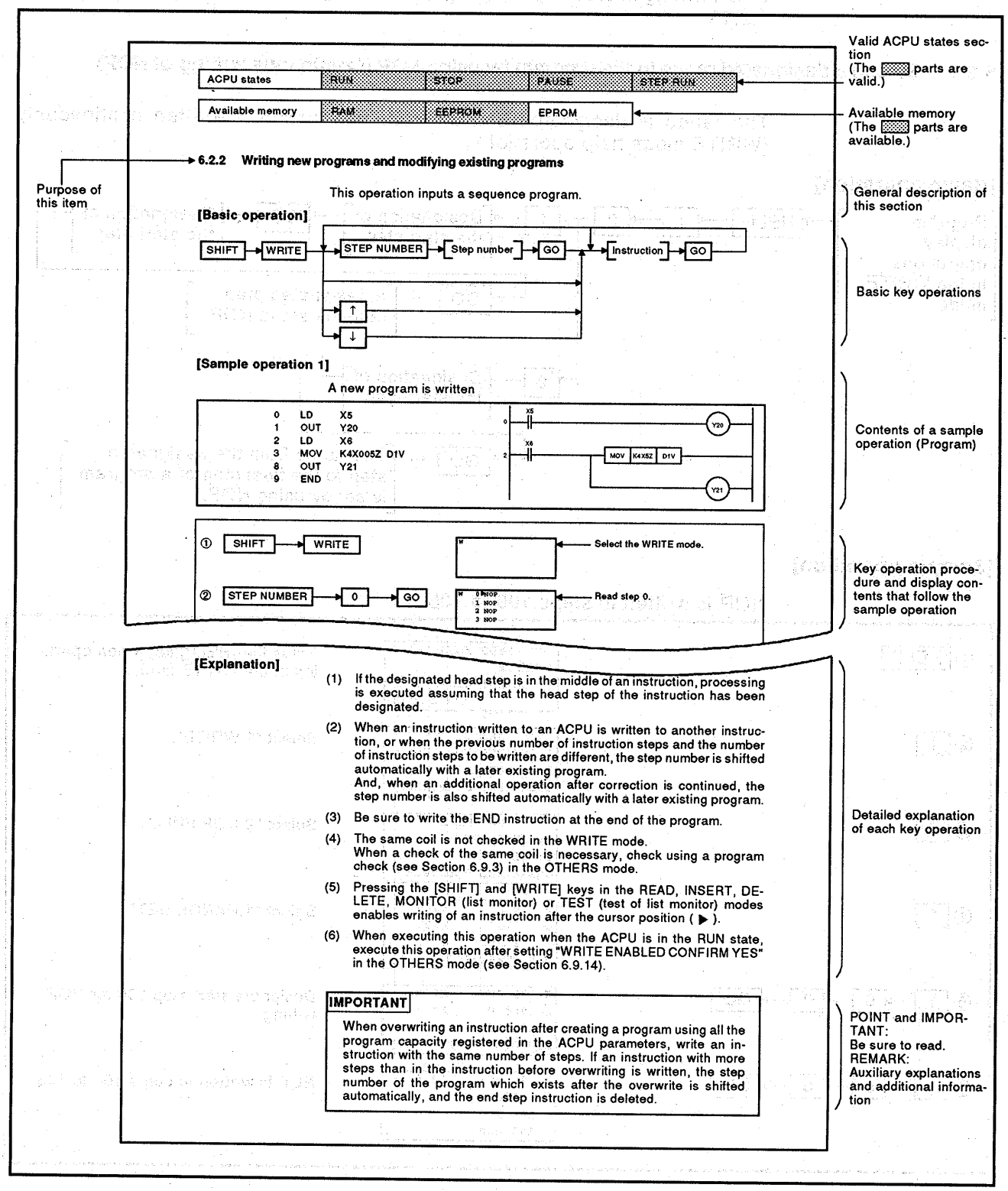
Set the PC numbers to be accessed in the OTHERS mode (see Section 6.9.10).

# 6. HOW TO USE EACH FUNCTION

This section explains how to use each function in all modes.

## 6.1 How to Understand the Operation Explanations

The explanation of each mode and the function explanations are configured as follows:


Valid ACPU states section (The ▨ parts are valid.)
ACPU states
RUN
STOP
PAUSE
STEP RUN
Available memory
RAM
EEPROM
EPROM
Available memory (The ▨ parts are available.)
Purpose of this item
6.2.2 Writing new programs and modifying existing programs
This operation inputs a sequence program.
General description of this section
[Basic operation]
SHIFT
WRITE
STEP NUMBER
Step number
GO
Instruction
GO
↑
↓
Basic key operations
[Sample operation 1]
A new program is written
0 LD X5
1 OUT Y20
2 LD X6
3 MOV K4X005Z D1V
8 OUT Y21
9 END
Contents of a sample operation (Program)
① SHIFT WRITE
Select the WRITE mode.
② STEP NUMBER 0 GO
Read step 0.
Key operation procedure and display contents that follow the sample operation
[Explanation]
(1) If the designated head step is in the middle of an instruction, processing is executed assuming that the head step of the instruction has been designated.
(2) When an instruction written to an ACPU is written to another instruction, or when the previous number of instruction steps and the number of instruction steps to be written are different, the step number is shifted automatically with a later existing program.
And, when an additional operation after correction is continued, the step number is also shifted automatically with a later existing program.
(3) Be sure to write the END instruction at the end of the program.
(4) The same coil is not checked in the WRITE mode.
When a check of the same coil is necessary, check using a program check (see Section 6.9.3) in the OTHERS mode.
(5) Pressing the [SHIFT] and [WRITE] keys in the READ, INSERT, DELETE, MONITOR (list monitor) or TEST (test of list monitor) modes enables writing of an instruction after the cursor position ( ▶ ).
(6) When executing this operation when the ACPU is in the RUN state, execute this operation after setting "WRITE ENABLED CONFIRM YES" in the OTHERS mode (see Section 6.9.14).
Detailed explanation of each key operation
IMPORTANT
When overwriting an instruction after creating a program using all the program capacity registered in the ACPU parameters, write an instruction with the same number of steps. If an instruction with more steps than in the instruction before overwriting is written, the step number of the program which exists after the overwrite is shifted automatically, and the end step instruction is deleted.
POINT and IMPORTANT: Be sure to read.
REMARK: Auxiliary explanations and additional information

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

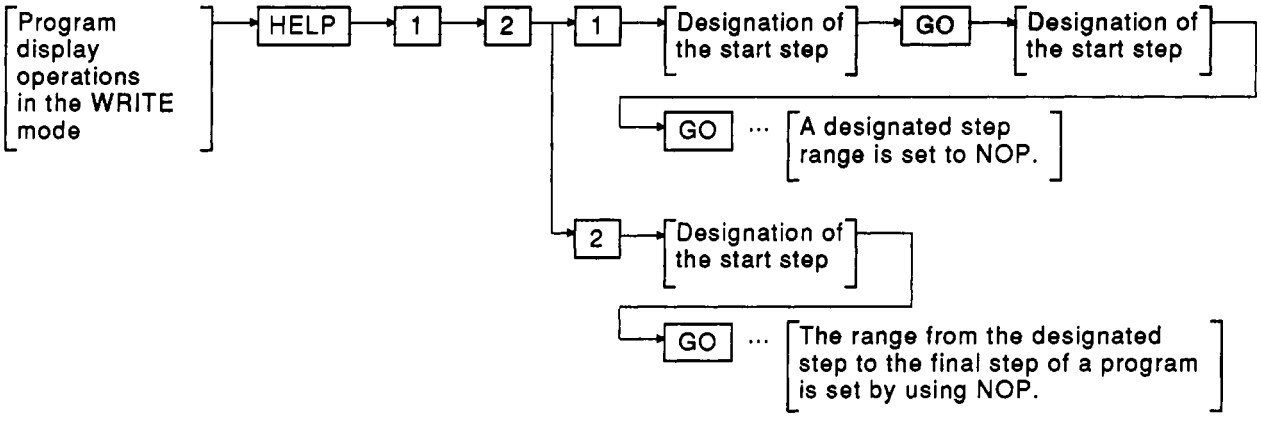
## 6.2 WRITE Mode (W) Operations

This mode writes programs to the ACPU and makes corrections (modifications to and insertions in) of ACPU programs.
When writing microcomputer programs, follow the procedures given in Section 6.9.13.

### 6.2.1 Setting a designated range in the program by using NOP (Continuous writing of NOP)

The range is designated and the NOP instruction is written continuously. (WRITE mode help operation)

**[Basic operation]**


Program display operations in the WRITE mode
HELP
1
2
1
Designation of the start step
GO
Designation of the start step
GO
... A designated step range is set to NOP.
2
Designation of the start step
GO
... The range from the designated step to the final step of a program is set by using NOP.

**[Sample operation]**

NOP is written to steps 100 to 105.

| | | |
|---|---|---|
| ① HELP | *** HELP MENU ***<br>1:WRITE<br>2:COMMENT DISPLAY<br>CLEAR:END | Press the [HELP] key when operating in the WRITE mode. |
| ② 1 | *** WRITE ***<br>1:INSTRUCTION HELP<br>2:NOP FILL<br>CLEAR:END HELP:MENU | Select "1:WRITE". |
| ③ 2 | NOP FILL<br>1:RANGE SET<br>2:ALL<br>CLEAR:END HELP:MENU | Select "2:NOP FULL". |
| ④ 1 | W 0 LD X000<br>1 OR X020<br>2 ANI M100<br>NOP START:STEP[ 1] | Select "1:RANGE SET". |
| ⑤ 1 → 0 → 0 → GO | W 99 OUT M200<br>100 LD X021<br>101 ANI M100<br>NOP END: STEP[ 100] | Designate start step 100 for NOP writing. |
| ⑥ 1 → 0 → 5 → GO | W 99 OUT X200<br>100 NOP<br>101 NOP<br>102 NOP | NOP is written to steps 100 to 105. |

**[Explanation]**

(1) When executing NOP batch writing by "1: RANGE SET", designate the start step and end step in the range to be written.
NOP is written to the range from the designated start step to the end step.

(2) When executing NOP batch writing by "2: ALL", designate the start step in the range to be written.
NOP is written to the range from the designated start step to the end step of the program.

(3) The start step and end step can be designated by using the [↑] and [↓] keys to move the cursor to the step to be designated and press [GO] or inputting the numbers.

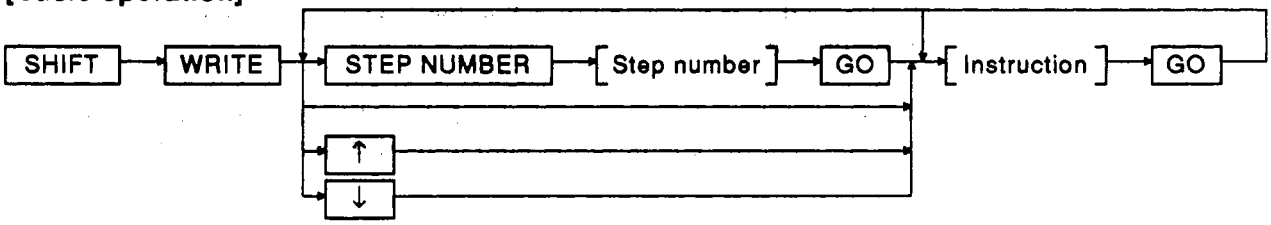
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.2.2 Writing new programs and modifying existing programs

This operation inputs a sequence program.

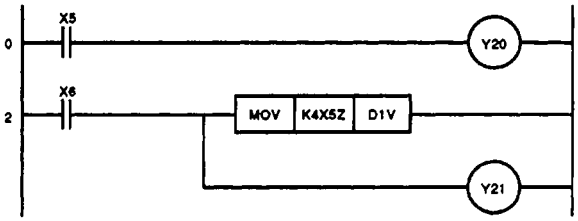
**[Basic operation]**




**[Sample operation 1]**

A new program is written

```
0    LD    X5
1    OUT   Y20
2    LD    X6
3    MOV   K4X005Z D1V
8    OUT   Y21
9    END
```




① SHIFT → WRITE

```
N
```

← Select the WRITE mode.

② STEP NUMBER → 0 → GO

```
N  0 NOP
   1 NOP
   2 NOP
   3 NOP
```

← Read step 0.

③ LD → X → 5 → GO

```
N  0 LD    X005
   1 NOP
   2 NOP
   3 NOP
```

④ OUT → Y → 2 → 0 → GO

```
N  0 LD    X005
   1 OUT   Y020
   2 NOP
   3 NOP
```

⑤ LD → X → 6 → GO

```
N  1 OUT   Y020
   2 LD    X006
   3 NOP
   4 NOP
```

⑥ MOV → K → 4 → X → 5 → Z → SP → D → 1 → V → GO

```
N  3  K4X005Z
   3  D1V
   8 NOP
   9 NOP
```

⑦ OUT → Y → 2 → 1 → GO

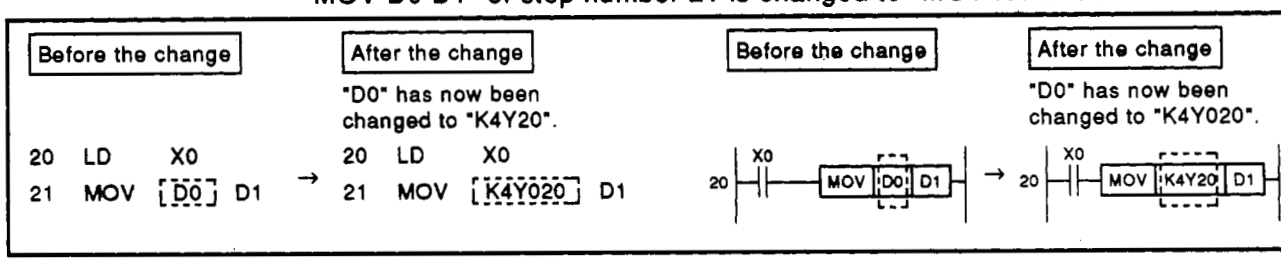
```
N  3  D1V
   8 OUT   Y021
   9 NOP
  10 NOP
```

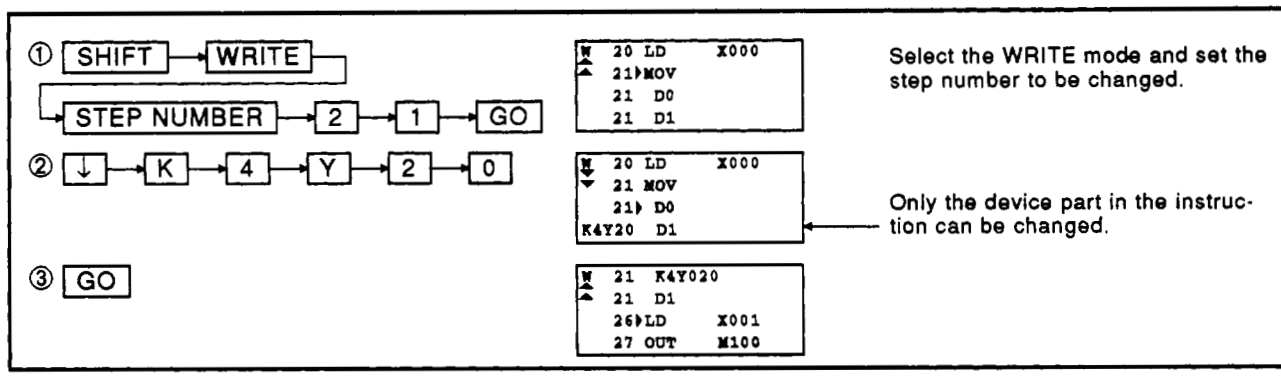
⑧ END → GO

```
N  8 OUT   Y021
   9 END
  10 NOP
  11 NOP
```

**[Sample operation 2]**

"MOV D0 D1" of step number 21 is changed to "MOV K4 Y100 K10"




**[Explanation]**

(1) If the designated head step is in the middle of an instruction, processing is executed assuming that the head step of the instruction has been designated.

(2) When an instruction written to an ACPU is written to another instruction, or when the previous number of instruction steps and the number of instruction steps to be written are different, the step number is shifted automatically with a later existing program.
And, when an additional operation after correction is continued, the step number is also shifted automatically with a later existing program.

(3) Be sure to write the END instruction at the end of the program.

(4) The same coil is not checked in the WRITE mode.
When a check of the same coil is necessary, check using a program check (see Section 6.9.3) in the OTHERS mode.

(5) Pressing the [SHIFT] and [WRITE] keys in the READ, INSERT, DELETE, MONITOR (list monitor) or TEST (test of list monitor) modes enables writing of an instruction after the cursor position ( ► ).

(6) When executing this operation when the ACPU is in the RUN state, execute this operation after setting "WRITE ENABLED CONFIRM YES" in the OTHERS mode (see Section 6.9.14).

**IMPORTANT**

When overwriting an instruction after creating a program using all the program capacity registered in the ACPU parameters, write an instruction with the same number of steps. If an instruction with more steps than in the instruction before overwriting is written, the step number of the program which exists after the overwrite is shifted automatically, and the end step instruction is deleted.

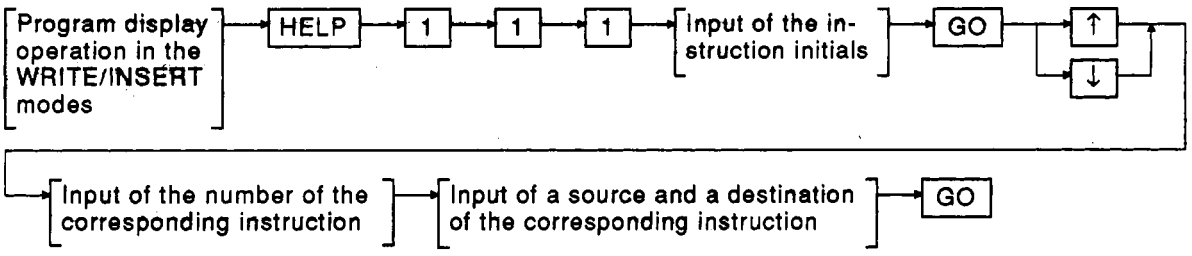
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.2.3 Displaying/selecting an instruction

This operation inputs the initials of an instruction and displays the instruction name. After selecting the displayed instruction name, writing/insertion can be operated continuously. (Help operation in the WRITE/INSERT modes)

**[Basic operation]**




**[Sample operation]**

The instruction name (RCL) whose initial is "R" is selected from the list and is written

| | | |
|---|---|---|
| ① | W 8 OUT Y021<br>▲ 9 LD X00C<br>▲ 10 NOP<br>11 NOP | When 16 bits of left rotation instruction data is written. |
| ② HELP | *** HELP MENU ***<br>1:WRITE<br>2:COMMENT DISPLAY<br>CLEAR:END | Press the [HELP] key when operating in the WRITE mode. |
| ③ 1 | *** WRITE ***<br>1:INSTRUCTION HELP<br>2:NOP FILL<br>CLEAR:END HELP:MENU | Select "1:WRITE". |
| ④ 1 | INSTRUCTION HELP<br>1:INSTR. SELECTION<br>2:STEP READ<br>CLEAR:END HELP:MENU | Select "1:INSTRUCTION HELP". |
| ⑤ 1 | *INSTR. SELECTION*<br>KEY IN FIRST LETTER<br>[ ]<br>CLEAR:END HELP:MENU | Select "1:INSTR. SELECTION". |
| ⑥ R → GO | *INSTR. SELECTION*<br>1 RCL N<br>2 RCR N<br>3 RET | The relationship between the instruction names that begin with "R", the source, and the destination are displayed. |
| ⑦ 1 → (←) → P → SP → K → 3 → GO | W 10 RCLP<br>▲ 10 K3<br>13 NOP<br>14 NOP | Select "1 RCL" from the instruction display list, and write "RCLP K3" at the cursor position. |

**[Explanation]**

(1) All A-series instructions are displayed.
Sometimes instructions that cannot be used with the ACPU to which the A8PUE is connected are displayed. The Programming Manual gives details about usable instructions.

(2) Dedicated instructions of an A2A and A3ACPU cannot be displayed.

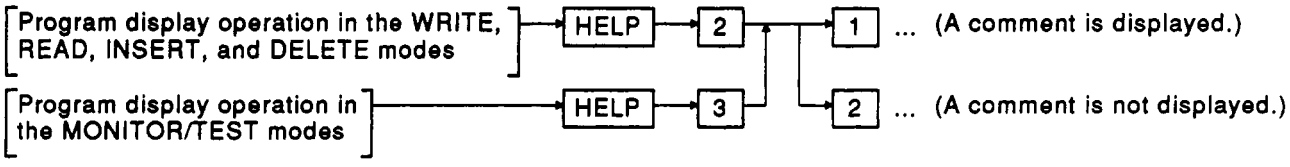
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.2.4 Comment display

This operation displays comment stored in an ACPU. (Help operation in the WRITE, READ, INSERT, DELETE, MONITOR, and TEST modes)

**[Basic operation]**




**[Sample operation]**

A comment display is set in the READ mode

① [HELP]

```
*** HELP MENU ***
1:WRITE
2:COMMENT DISPLAY
          CLEAR:END
```

Press the [HELP] key when operating in the READ mode.

② [2]

```
  COMMENT DISPLAY
1:YES
2:NO
CLEAR:END HELP:MENU
```

Select "2:COMMENT DISPLAY".

③ [1]

```
R   0 LD    X000
▲   1 OUT   Y020
    2 LD    X000
  MOTOR START LS
```

Select "1:YES".
The comment that is attached to the device at the cursor position is displayed.

④ [↓]

```
R   0 LD    X000
▲   1 OUT   Y020
    2 LD    X000
  MOTOR START CNFRM
```

**[Explanation]**

(1) Comments cannot be created in the A8PUE.
To display a comment, create the comment by using the GPP function of a peripheral device, and then write it to an ACPU.

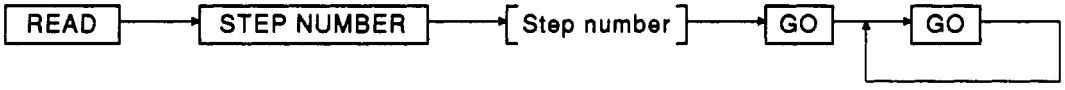
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | EPROM | |

## 6.3 READ Mode (R) Operations

This mode reads programs written to an ACPU by designating step numbers, instructions, or devices.
Section 6.9.13 explains how to read microcomputer programs.

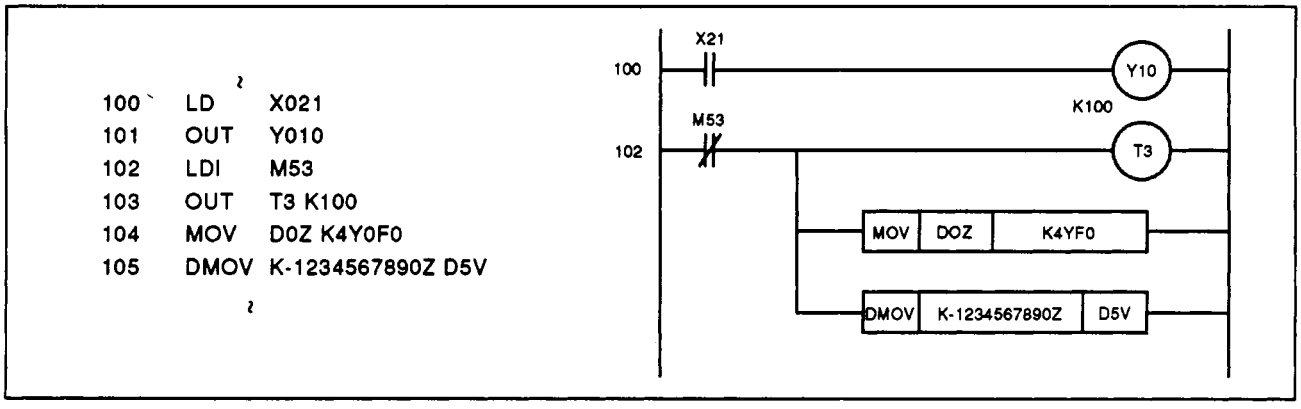
### 6.3.1 Reading the instruction of the step number designated in the program. (Instruction read by designating the step number)

**[Basic operation]**

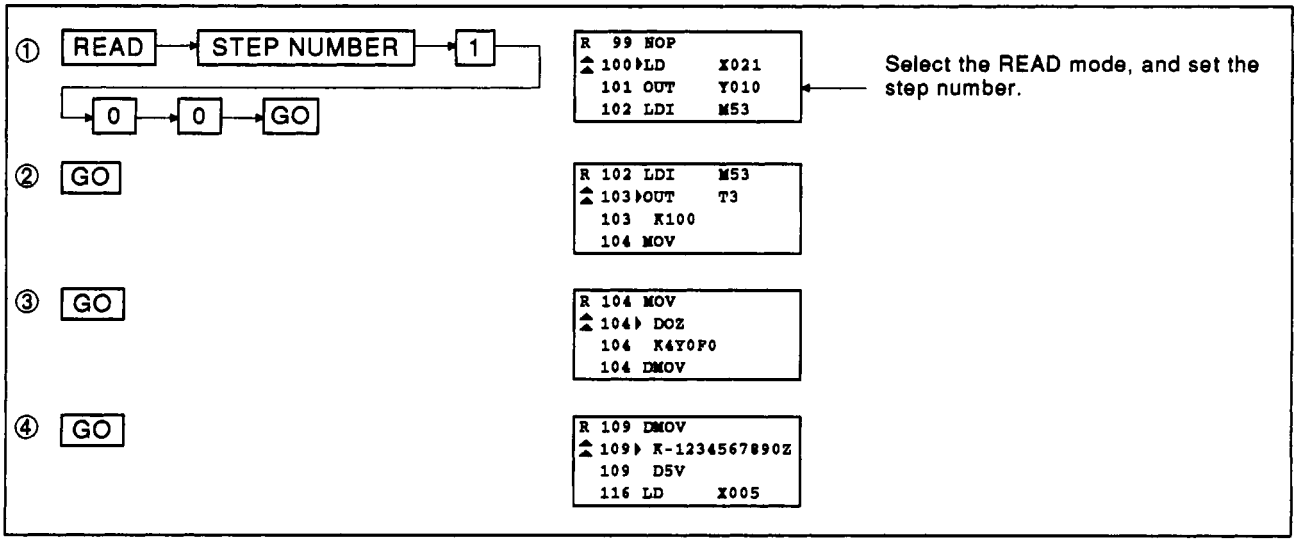



**[Sample operation]**

The instruction that was designated after step number 100 in the following program is read.




**[Sample operation]**



**[Explanation]**

(1) If the designated head step is in the middle of an instruction, processing is executed assuming that the head step of the instruction has been designated.
(Example) When step 106 is designated in the example on the previous page, steps 104 and after are displayed.

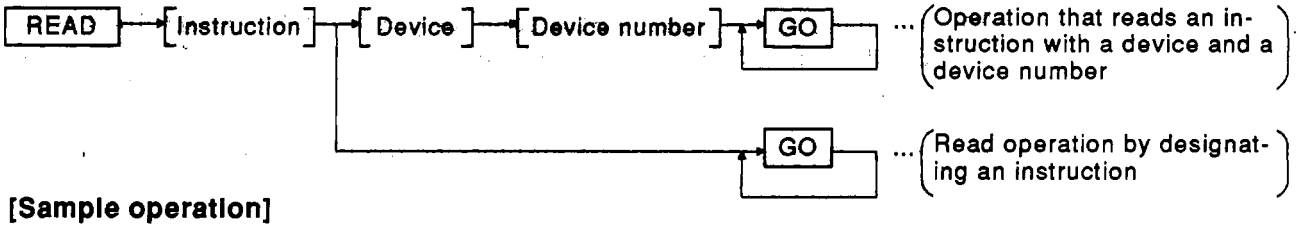
(2) When the ACPU is in the RUN state, processing takes longer than in the STOP state. Therefore, wait until an instruction to be read is displayed. (The processing time is proportional to the number of steps in a program.)

(3) If the [GO] key is pressed after pressing the [STEP NUMBER], the step number, and the [GO] keys, the screen is scrolled up.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | EPROM | |

### 6.3.2 Reading an instruction designated in the program (Instruction read by designating an instruction)

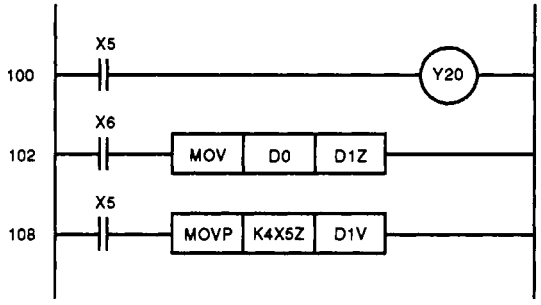
**[Basic operation]**




**[Sample operation]**

Instruction of step number where LD X5 or MOV instruction is used in the program is read.

```
100   LD    X005
101   OUT   Y020
102   LD    X006
103   MOV   D0 D1Z
108   LD    X005
109   MOVP  K4X005Z D1V
```




① READ → LD → X → 5

```
R   5 OUT    M100
  6 OUT    M200
    7 LD     X000
LD X5
```

Select the READ mode and set the instruction, device, and device number of the instruction to be searched.

② GO

```
R  99 OUT    M200
100 LD     X005
  101 OUT    Y020
  102 LD     X006
```

Read the 100th step "LD X005".

③ GO

```
R 103  D1Z
108 LD     X005
  109 MOVP
  109  K4X005Z
```

Read the 108th step "LD X005".

④ GO

```
R 103  D1Z
108 LD     X005
  109 MOVP
NOT FOUND
```

Message when there is no "LD X005" instruction after step 109

⑤ CLEAR → MOV

```
R 103  D1Z
108 LD     X005
  109 MOVP
MOV
```

Set only the instructions to be read.

⑥ GO

```
R 102 LD     X006
103 MOV
  103  D0
  103  D1Z
```

Read the 103rd step "MOV D0 D1Z".

⑦ GO

```
R 108 LD     X005
109 MOVP
  109  K4X005Z
  109  D1V
```

Read the 109th step "MOVP K4X005Z D1V".

⑧ GO

```
R 108 LD     X005
109 MOVP
  109  K4X005Z
NOT FOUND
```

Message when there is no "MOV" instruction after step 114

**[Explanation]**

(1) The instruction is searched from step 0 by pressing the [GO] key after setting an instruction. And then, the detected step is displayed. Thereafter, an instruction is searched from the following step by pressing the [GO] key, and the detected step is displayed.
When an instruction cannot be detected even if it is searched to the END instruction, PROGRAM NOT FOUND is displayed.

(2) The method of inputting the instruction to be designated by this operation is as follows.

1) Designate the following instruction with a device and a device number. (MC and MCR can only be designated by an instruction.)
LD OR AND SET PLS SFT SFTP OUT MC
LDI ORI ANI RST MCR

(a) Designate the OUT instruction with the first device.
(Example)
LD X5
OUT T123 (OUT T123 K123 is impossible.)
MC N3 or MC (MC N3 M1023 is impossible.)

(b) When an A2A or A3ACPU is used, be sure to modify indexes to the device number when reading an instruction modified with the indexes Z, Z1 to Z6, V and V1 to V6 to the device number.

(Example)

| Designated Instructions | Instructions to be Read |
|---|---|
| LD X5 | LD X5 only |
| LD X5V | LD X5V only |

2) P and I instructions cannot be read by this operation.
Follow the operation given in Section 6.3.3.

3) For instructions other than the above, designate instructions only.

(3) Additional explanations of comparison and pulse ( [ Instruction ] P) instructions.
The following related instructions can be also read by a designated instruction by using this operation.
When restricting instructions to be read, designate all instructions. (For example, LD = and LDD =, etc.)

1) Comparison instructions

| Designated Instructions | Instructions to be Read |
|---|---|
| = | LD= , AND= , OR= |
| D= | LDD= , ANDD= , ORD= |
| <> | LD<> , AND<> , OR<> |
| D<> | LDD<> , ANDD<> , ORD<> |
| > | LD> , AND> , OR> |
| D> | LDD> , ANDD> , ORD> |
| <= | LD<= , AND<= , OR<= |
| D<= | LDD<= , ANDD<= , ORD<= |
| < | LD< , AND< , OR< |
| D< | LDD< , ANDD< , ORD< |
| >= | LD>= , AND>= , OR>= |
| D>= | LDD>= , ANDD>= , ORD>= |

2) Pulse instructions

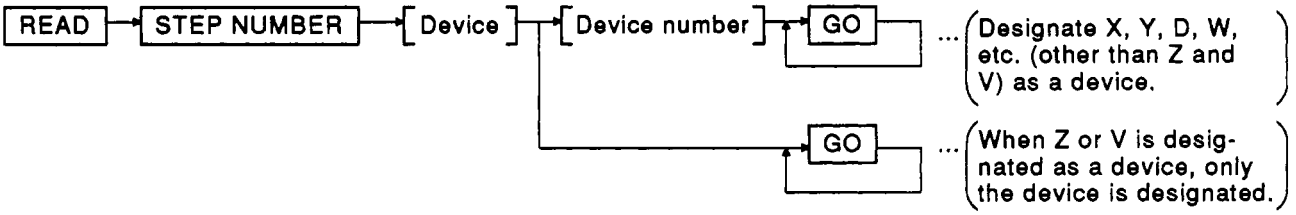
| Designated Instructions | Instructions to be Read | Examples |
|---|---|---|
| [ Instruction ] | [ Instruction ] and [ Instruction ] P | MOV, MOVP |
| [ Instruction ] P | [ Instruction ] P | MOVP |

(4) Changing input data
Before pressing the [GO] key, data can be changed by the following operations:
- After pressing the [CLEAR] key, reenter the correct data.
- When changing an [ Instruction ] , reenter the instruction after pressing the [CLEAR] key.
- When changing a [ device ] and a [ device number ] , reenter the correct data.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | EPROM | |

### 6.3.3 Reading an instruction by designating a used device in the program (Instruction read by designating a device)

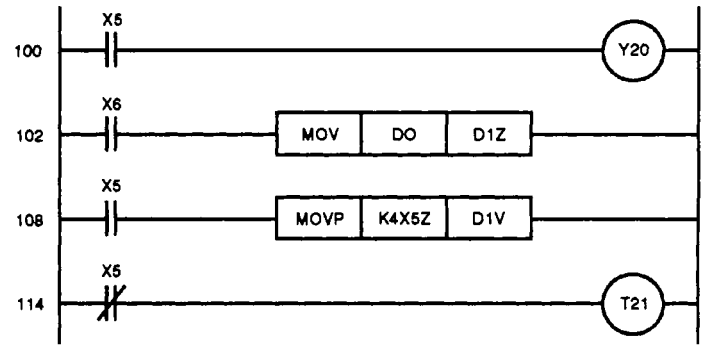
**[Basic operation]**

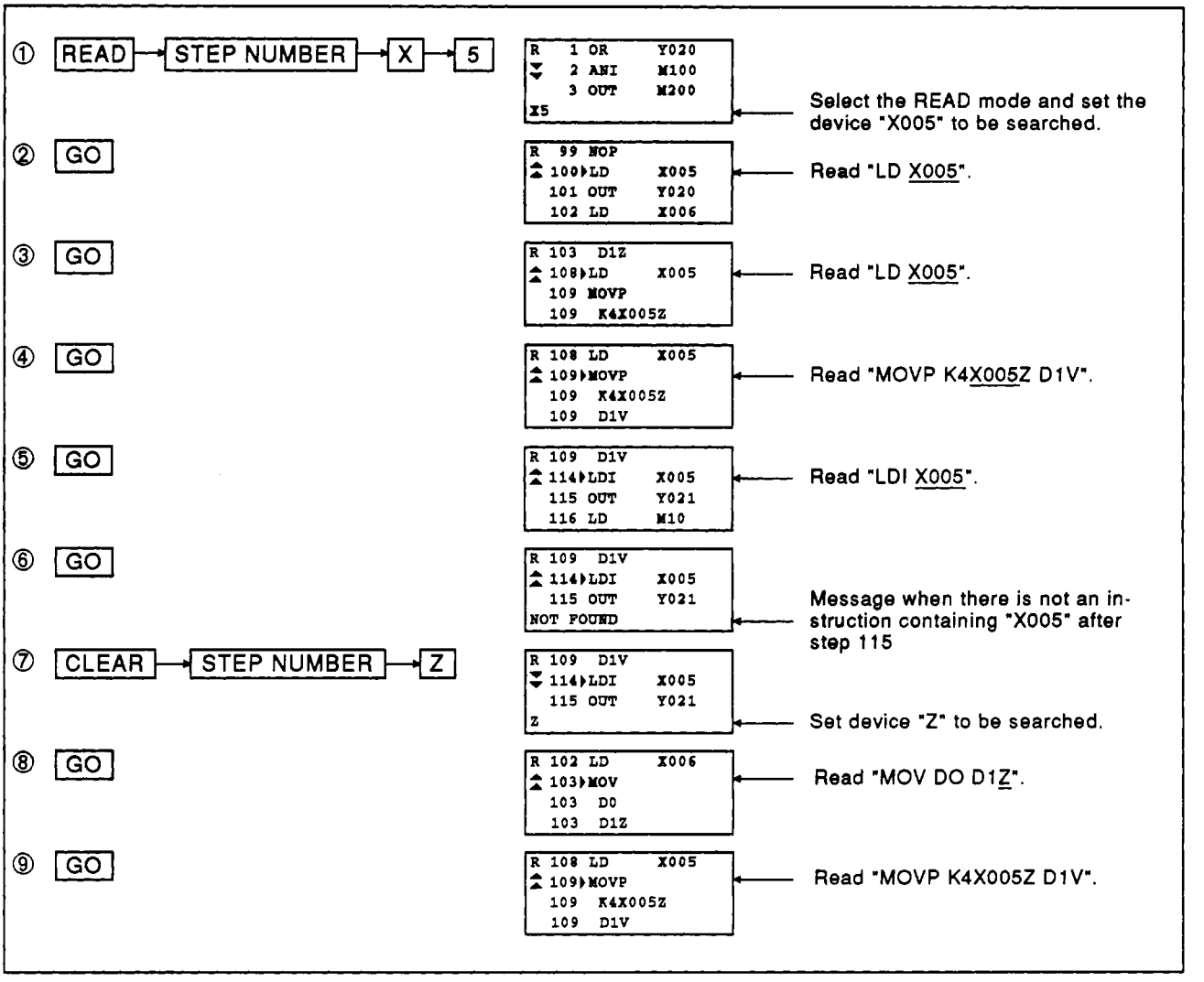

READ
STEP NUMBER
Device
Device number
GO
GO
... Designate X, Y, D, W, etc. (other than Z and V) as a device.
... When Z or V is designated as a device, only the device is designated.

**[Sample operation]**

The instruction of the step number for which devices "X5" and "Z" are used is read in the following program

```
100   LD    X005
101   OUT   Y020
102   LD    X006
103   MOV   D0 D1Z
108   LD    X005
109   MOVP  K4X005Z D1V
114   LDI   X005
115   OUT   Y021
```



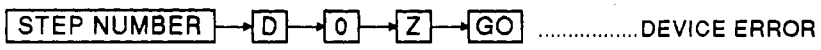

① READ → STEP NUMBER → X → 5
R 1 OR Y020
2 ANI M100
3 OUT M200
X5
Select the READ mode and set the device "X005" to be searched.
② GO
R 99 NOP
100 LD X005
101 OUT Y020
102 LD X006
Read "LD X005".
③ GO
R 103 D1Z
108 LD X005
109 MOVP
109 K4X005Z
Read "LD X005".
④ GO
R 108 LD X005
109 MOVP
109 K4X005Z
109 D1V
Read "MOVP K4X005Z D1V".
⑤ GO
R 109 D1V
114 LDI X005
115 OUT Y021
116 LD M10
Read "LDI X005".
⑥ GO
R 109 D1V
114 LDI X005
115 OUT Y021
NOT FOUND
Message when there is not an instruction containing "X005" after step 115
⑦ CLEAR → STEP NUMBER → Z
R 109 D1V
114 LDI X005
115 OUT Y021
Z
Set device "Z" to be searched.
⑧ GO
R 102 LD X006
103 MOV
103 D0
103 D1Z
Read "MOV D0 D1Z".
⑨ GO
R 108 LD X005
109 MOVP
109 K4X005Z
109 D1V
Read "MOVP K4X005Z D1V".

**[Explanation]**

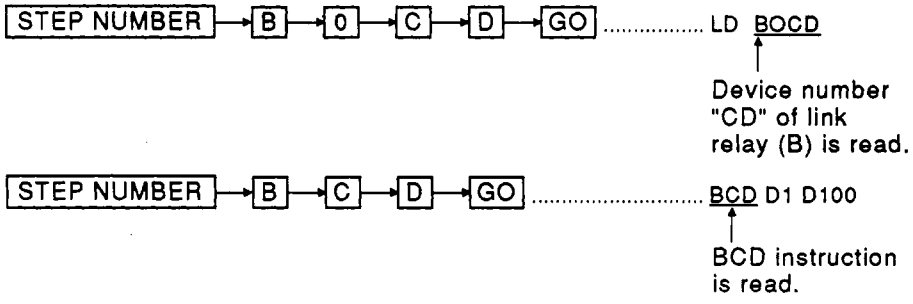
(1) The device and device number are searched from step 0 by pressing the [GO] key after setting device and device number. And then, the detected step is displayed. Thereafter, an instruction is searched from the following step by pressing the [GO] key, and the detected step is displayed. When an instruction cannot be detected even if it is searched to the END instruction, PROGRAM NOT FOUND is displayed.

- Changing input data Before pressing the [GO] key, data can be changed by the following operations:
  - After pressing the [CLEAR] key, reenter the correct data.
  - When changing a [ device ] and a [ device number ] , reenter the correct data.

(2) When reading by using device numbers, a designated device modified by indexes Z or V (when an A2A or A3ACPU is used: Z, Z1 to Z6; V, V1 to V6) is also read as the corresponding device. (See sample operation 4).)
If a device number with indexes is designated, a DEVICE ERROR occurs and it cannot be read.
(Example)




(3) When reading device number "CD" of link relay (B), input [B], [0], [C] and [D]. If only [B] , [C] and [D] are input, a BCD instruction is read.
(Example)

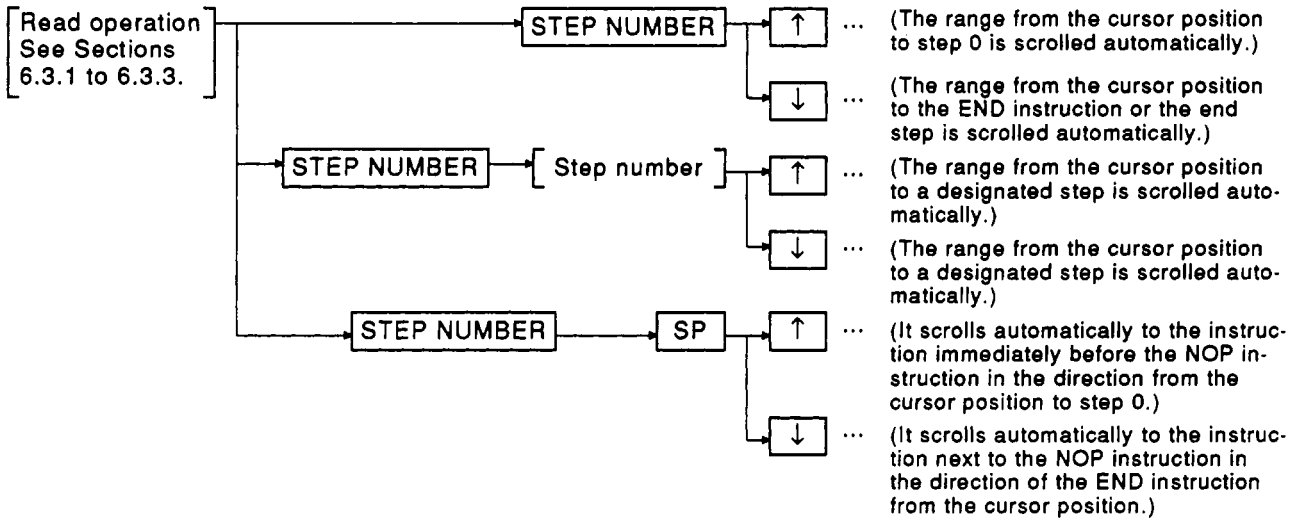


| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

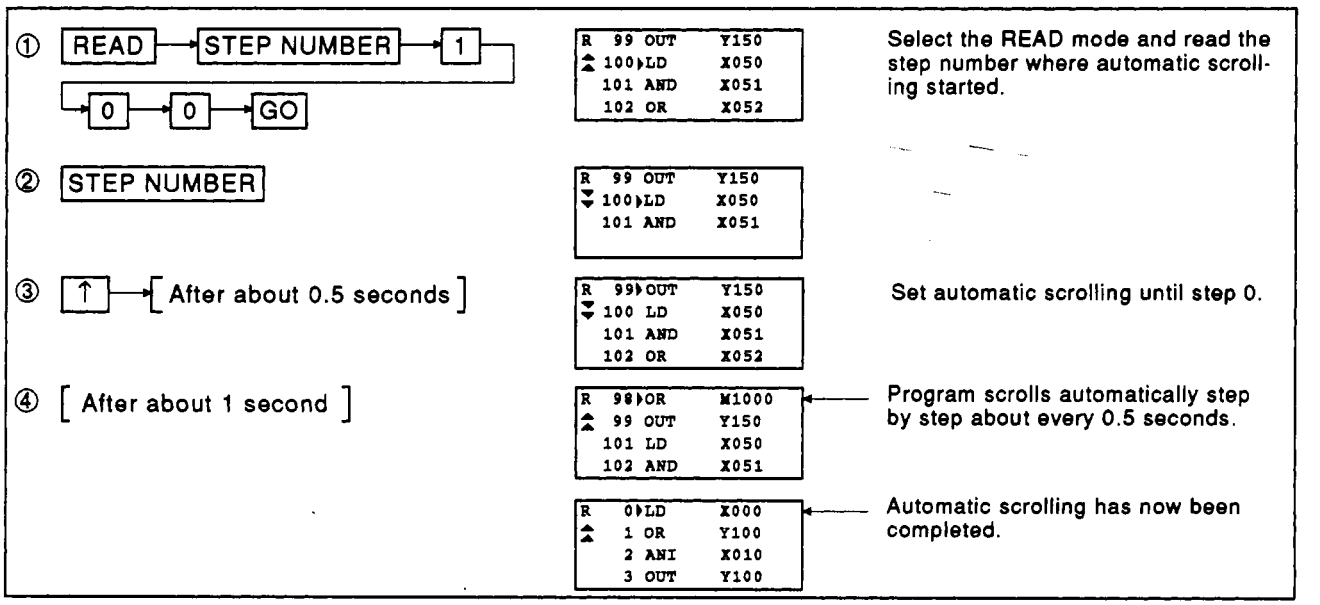
### 6.3.4 Automatic scrolling of a program

**[Basic operation]**


Read operation
See Sections
6.3.1 to 6.3.3.
STEP NUMBER
↑
(The range from the cursor position to step 0 is scrolled automatically.)
↓
(The range from the cursor position to the END instruction or the end step is scrolled automatically.)
STEP NUMBER
Step number
↑
(The range from the cursor position to a designated step is scrolled automatically.)
↓
(The range from the cursor position to a designated step is scrolled automatically.)
STEP NUMBER
SP
↑
(It scrolls automatically to the instruction immediately before the NOP instruction in the direction from the cursor position to step 0.)
↓
(It scrolls automatically to the instruction next to the NOP instruction in the direction of the END instruction from the cursor position.)

**[Sample operation]**

A program is scrolled automatically to step 0


① READ STEP NUMBER 1 0 0 GO
R 99 OUT Y150
100 LD X050
101 AND X051
102 OR X052
Select the READ mode and read the step number where automatic scrolling started.
② STEP NUMBER
R 99 OUT Y150
100 LD X050
101 AND X051
③ ↑ After about 0.5 seconds
R 99 OUT Y150
100 LD X050
101 AND X051
102 OR X052
Set automatic scrolling until step 0.
④ After about 1 second
R 98 OR M1000
99 OUT Y150
101 LD X050
102 AND X051
Program scrolls automatically step by step about every 0.5 seconds.
R 0 LD X000
1 OR Y100
2 ANI X010
3 OUT Y100
Automatic scrolling has now been completed.

**[Explanation]**

(1) The program scrolls automatically from the cursor position to the direction set by [ ↑ ] or [ ↓ ] keys.

(2) Press any key to interrupt scrolling.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

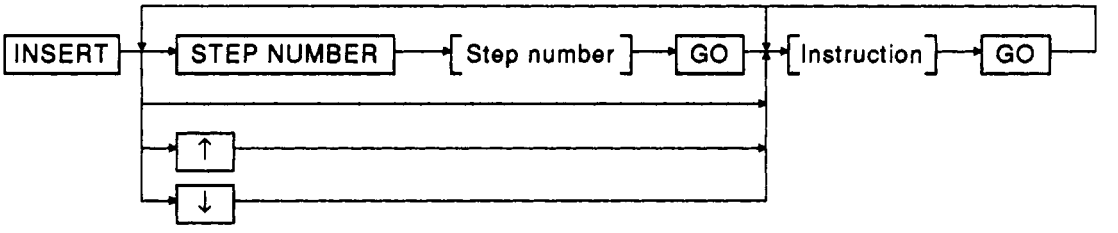
| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

## 6.4 INSERT (I) Mode Operations

This mode inserts new instructions to modify ACPU programs.

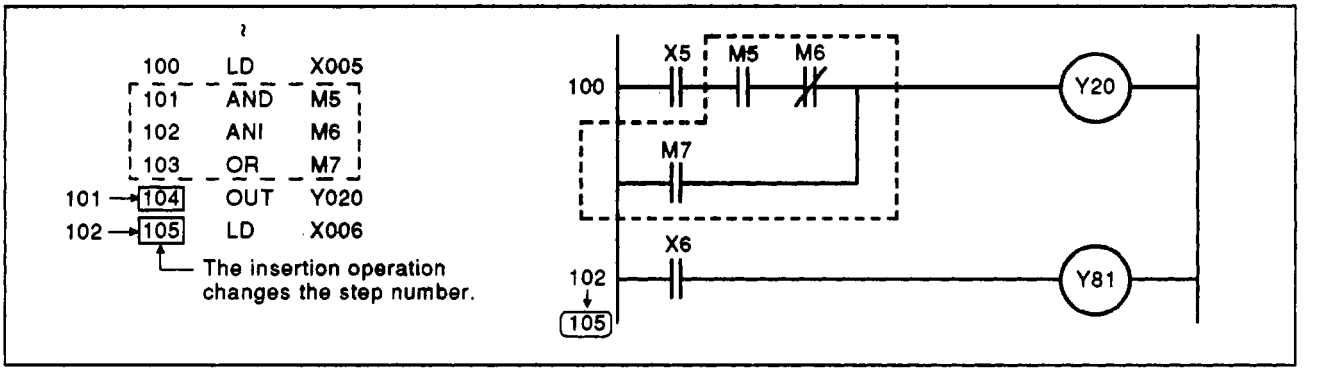
### 6.4.1 Inserting an instruction in a program (Program insertion)

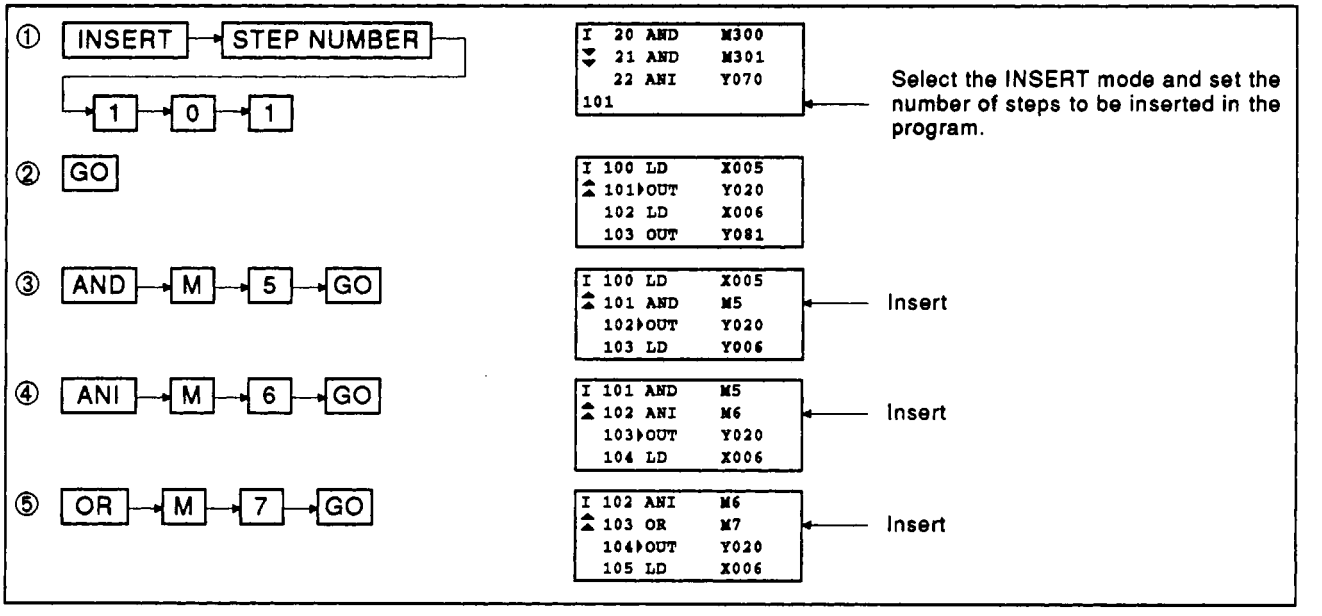
**[Basic operation]**




**[Sample operation]**

A program (dotted area) is inserted

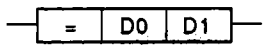


**[Explanation]**

(1) When this operation is completed normally, the inserted instruction is written to an ACPU as it is, and the following step number is displayed.

(2) Instructions must not be input by step units; use instruction units.
(Example)

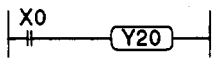



Do not insert AND=, D0 and D1 separately. Insert AND=, D0, and D1 together.
After the operation has been completed, confirm an inserted part in the READ mode.

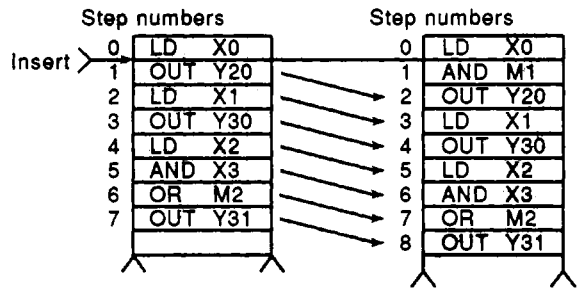
(3) If the designated head step is in the middle of an instruction, processing is executed assuming that the head step of the instruction has been designated.

(4) When an instruction is inserted, the step number is shifted automatically in the existing program.

1)




When AND M1 is inserted in Y20




(5) The same coil is not checked in the INSERT mode.
When a check of the same coil is necessary, check using a program check (see Section 6.9.3) in the OTHERS mode.

(6) Pressing the [INSERT] key in the READ, INSERT, DELETE, MONITOR (list monitor) or TEST (test of list monitor) modes enables inserting of an instruction after the cursor position (▶).

(7) After pressing the [STEP NUMBER] , step number, and [GO] keys, press the [GO] key to scroll the program up.

(8) When executing this operation when the ACPU is in the RUN state, execute this operation after setting "WRITE ENABLED CONFIRM YES" in the OTHERS mode (see Section 6.9.14).

**IMPORTANT**

When overwriting an instruction after creating a program using all the program capacity registered in the ACPU parameters, write an instruction with the same number of steps. If an instruction with more steps than in the instruction before overwriting is written, the step number of the program which exists after the overwrite is shifted automatically, and the end step instruction is deleted.

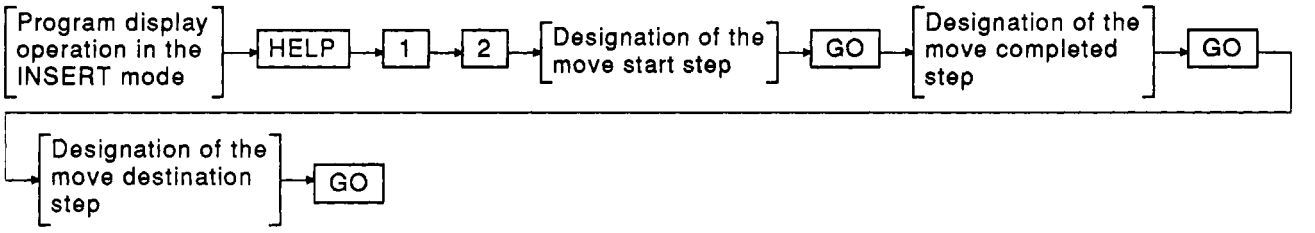
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.4.2 Batch moving a program

This operation moves a program in a designated range to a designated position. (Help operation in the INSERT mode)

**[Basic operation]**




**[Sample operation]**

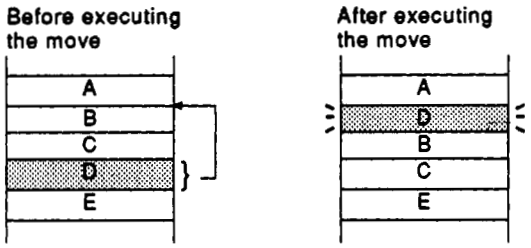
Steps 16 to 18 are moved to step 50

| | | |
|---|---|---|
| ① HELP | *** HELP MENU ***<br>1:INSERT<br>2:COMMENT DISPLAY<br>CLEAR:END | Press the [HELP] key when operating in the READ mode. |
| ② 1 | *** INSERT *** 1/2<br>1:INSTR. HELP<br>2:TRANSFER<br>CLEAR:END HELP:MENU | Select "1:INSERT". |
| ③ 2 → 1 → 6 | I 0▶LD X000<br>▼ 1 OUT Y020<br>▼ 2 LD X000<br>START STEP [ 16] | Select "2:TRANSFER".<br>Set 16 as the move start step. |
| ④ GO → 1 → 8 | I 15 OUT Y023<br>▼ 16▶LD Y023<br>▼ 17 ANI T1<br>END STEP [ 18] | Set 18 as the move END step. |
| ⑤ GO → 5 → 0 | I 17 ANI T1<br>▼ 18▶OUT T0<br>▼ 18 K10<br>DESTINATION [ 50] | Set 50 as the move destination step. |
| ⑥ GO | I 49 K10<br>▲ 50▶NOP<br>▲ 51 NOP<br>▲ 52 NOP | The program of steps 16 to 18 has been moved to just before step 50. |

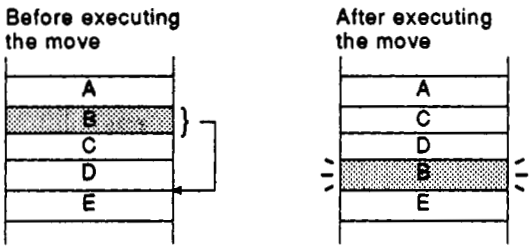
**[Explanation]**

(1) When the move is executed, it is moved to just before the move destination step.

(a) If the move destination is before a movement source, instructions in the range from the movement destination to the movement source are shifted in descending order.
The figure below is an example of shifting ladder block D before ladder block B.




(b) When a move destination is after a movement source, instructions in the range from the move destination to the movement source are shifted in ascending order.
The figure below is an example of shifting ladder block B before ladder block E.



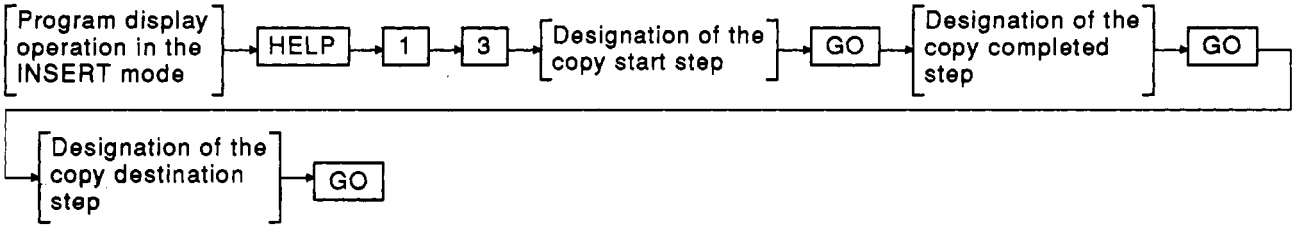
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.4.3 Copying a program

This operation copies a program to the designated position when it is composed of an instruction that is the same as the created program. (Help operation in the INSERT mode)

**[Basic operation]**



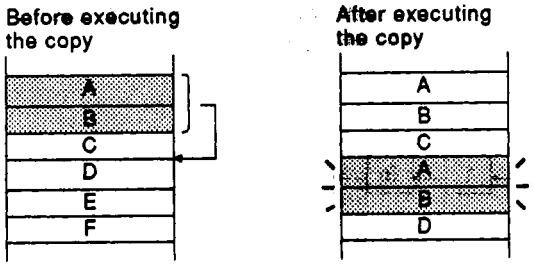

**[Sample operation]**

When steps 15 to 20 are copied to step 60

| | | |
|---|---|---|
| ① HELP | *** HELP MENU ***<br>1:INSERT<br>2:COMMENT DISPLAY<br>CLEAR:END | Press the [HELP] key when operating in the INSERT mode. |
| ② 1 | *** INSERT *** 1/2<br>1:INSTR. HELP<br>2:TRANSFER<br>CLEAR:END HELP:MENU | Select "1:INSERT". |
| ③ ↓ | *** INSERT *** 2/2<br>3:COPY<br><br>CLEAR:END HELP:MENU | Switch to the display "*** INSERT *** 2/2". |
| ④ 3 → 1 → 6 | I 0 LD X000<br>1 OUT Y020<br>2 LD X000<br>START STEP [ 16] | Select "3:COPY".<br>Set 16 as the copy start step. |
| ⑤ GO → 1 → 8 | I 15 OUT Y023<br>16 LD Y023<br>17 ANI T1<br>END STEP [ 18] | Set 18 as the copy END step. |
| ⑥ GO → 6 → 0 | I 17 ANI T1<br>18 OUT T0<br>18 K10<br>DESTINATION [ 60] | Set 60 as the copy destination step. |
| ⑦ GO | I 59 NOP<br>60 LD Y023<br>61 ANI T1<br>62 OUT T0 | The program of step 16 to 18 is copied to step 60 and after. |

**[Explanation]**

(1) When the copy is executed, a program is copied to just after the copy destination step, and the subsequent instructions are shifted back.
The figure below is an example of copying ladder blocks A and B before ladder block D.




(2) When pointer (P, I) is included in the range to be copied, SETTING ERROR occurs.
Leave out pointers of ranges to be copied.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

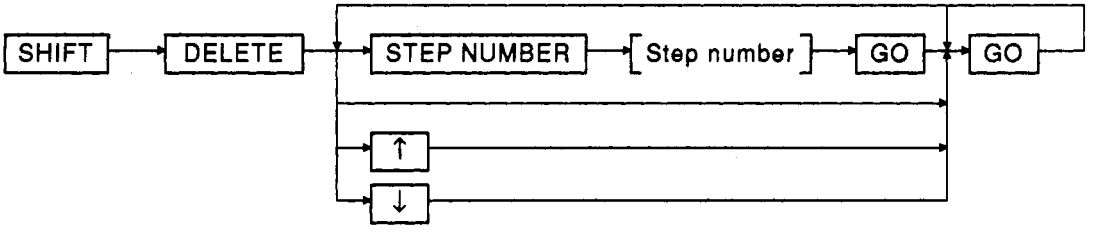
| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

## 6.5 DELETE (D) Mode Operations

This mode deletes existing instructions to modify ACPU programs.

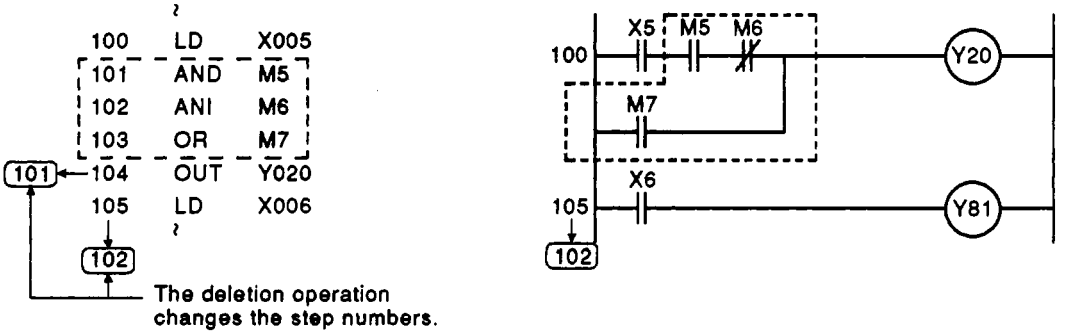
### 6.5.1 Deleting an instruction in a program (Program deletion)

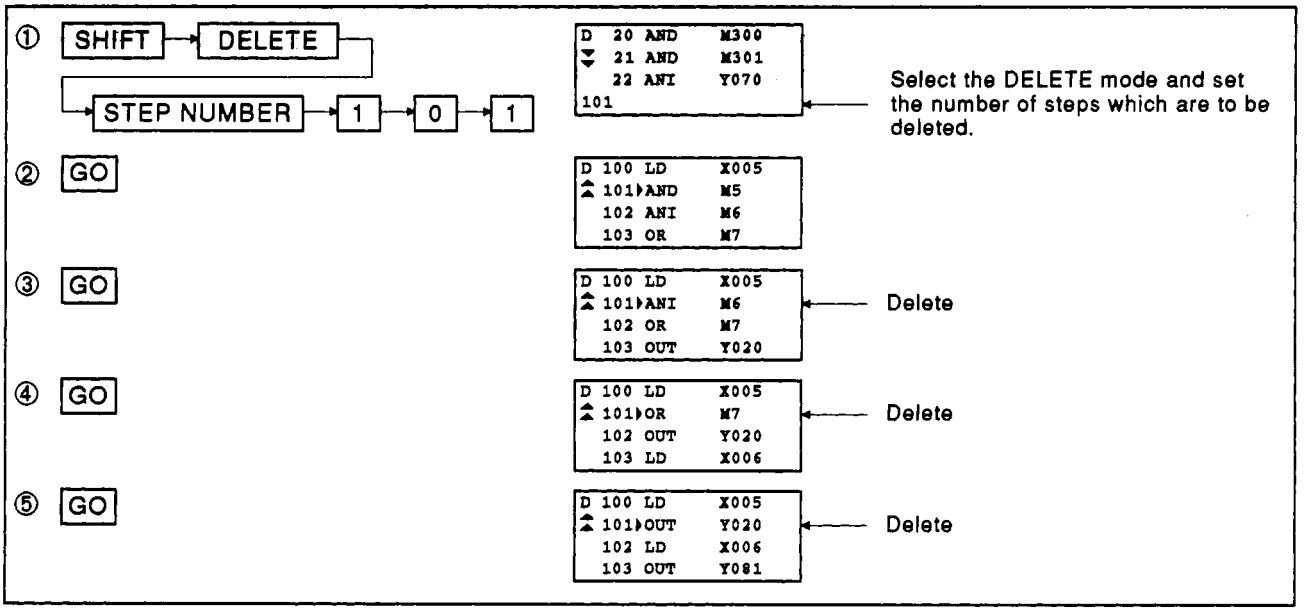
**[Basic operation]**




**[Sample operation]**

A program (dotted area) is deleted

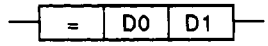


**[Explanation]**

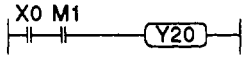
(1) If the designated head step is in the middle of an instruction, processing is executed assuming that the head step of the instruction has been designated.

(2) Instructions must not be deleted by step units; use instruction units.
(Example)

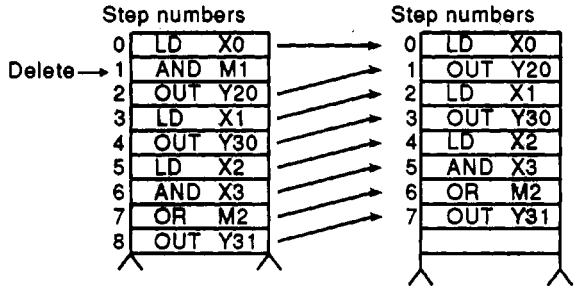



Do not delete AND=, D0 and D1 separately. Delete AND=, D0, and D1 together.
After the operation has been completed, confirm a deleted part in the READ mode.

(3) When the operation is completed normally, the instruction of a designated step number is deleted from an ACPU and the step number is now displayed with the next instruction.




When AND M1 of Y20 is deleted




(4) Pressing the [SHIFT] and [DELETE] keys in the READ, WRITE, INSERT, MONITOR (list monitor) or TEST (test of list monitor) modes enables deleting of an instruction after the cursor position (▶).

(5) When executing this operation when the ACPU is in the RUN state, execute this operation after setting "WRITE ENABLED CONFIRM YES" in the OTHERS mode (see Section 6.9.14).

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

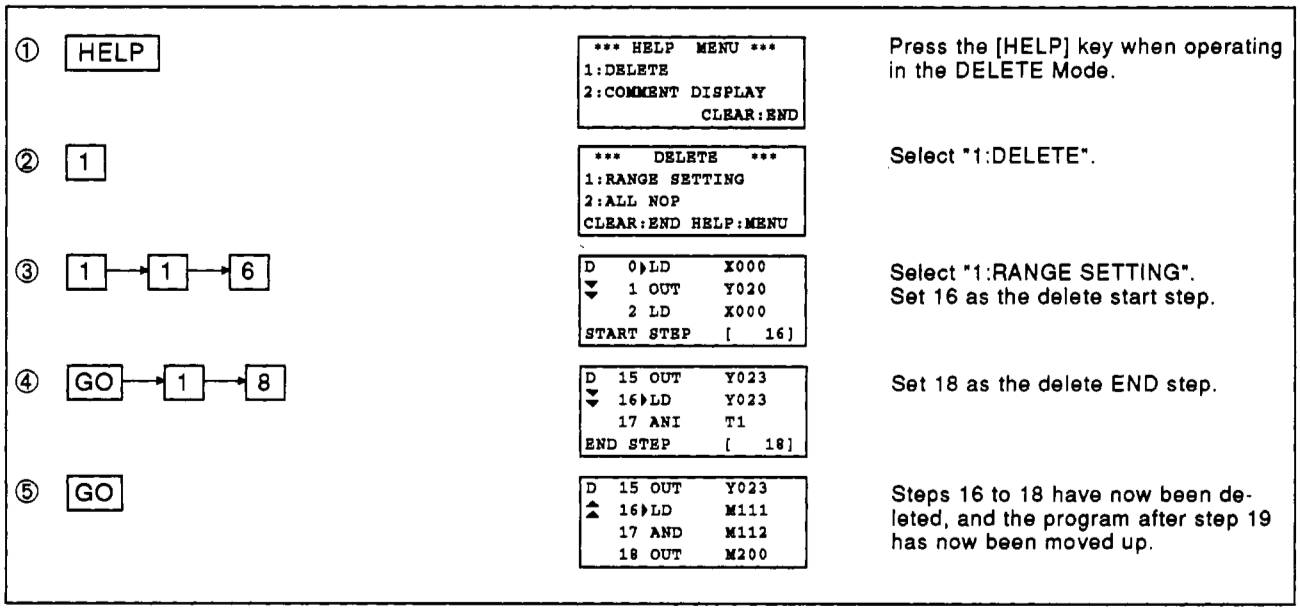
### 6.5.2 Deletion by designating a range

This operation deletes several instructions by designating a range. (Help operation in the DELETE mode)

**[Basic operation]**

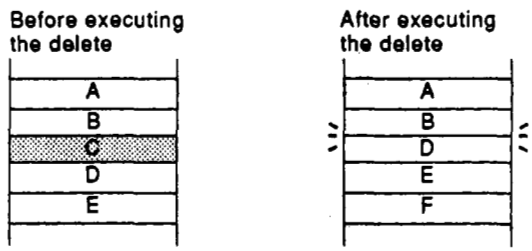
**[Sample operation]**

Steps 16 to 18 are deleted


① HELP
*** HELP MENU ***
1:DELETE
2:COMMENT DISPLAY
CLEAR:END
Press the [HELP] key when operating in the DELETE Mode.
② 1
*** DELETE ***
1:RANGE SETTING
2:ALL NOP
CLEAR:END HELP:MENU
Select "1:DELETE".
③ 1 → 1 → 6
D 0 LD X000
1 OUT Y020
2 LD X000
START STEP [ 16]
Select "1:RANGE SETTING". Set 16 as the delete start step.
④ GO → 1 → 8
D 15 OUT Y023
16 LD Y023
17 ANI T1
END STEP [ 18]
Set 18 as the delete END step.
⑤ GO
D 15 OUT Y023
16 LD M111
17 AND M112
18 OUT M200
Steps 16 to 18 have now been deleted, and the program after step 19 has now been moved up.

**[Explanation]**

(1) When deletion is executed, the program is moved up. The figure below is an example of deleting ladder block C.



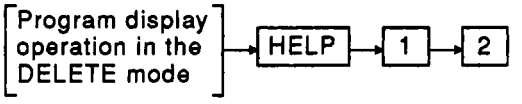
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.5.3 Batch deletion of NOP in the program

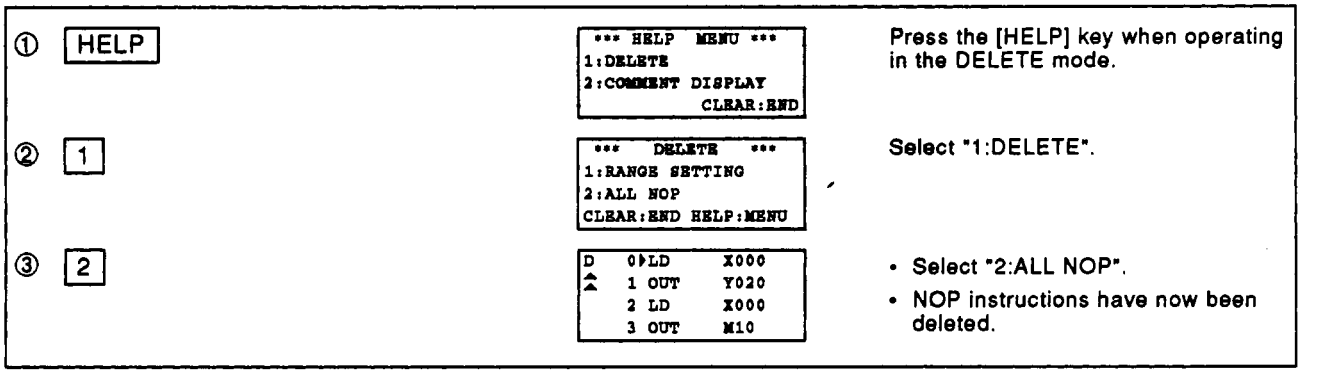
This operation deletes all NOP instructions in the program (step 0 to the END instruction). (Help operation in the DELETE mode)

**[Basic operation]**




**[Sample operation]**

NOP instructions are batch-deleted




**[Explanation]**

(1) When batch deletion of NOP instructions has been executed, the program is condensed.

(2) The NOPLF instruction (instruction used to change a page at any position when printing a circuit diagram and an instruction list using the GPP function) is not deleted.

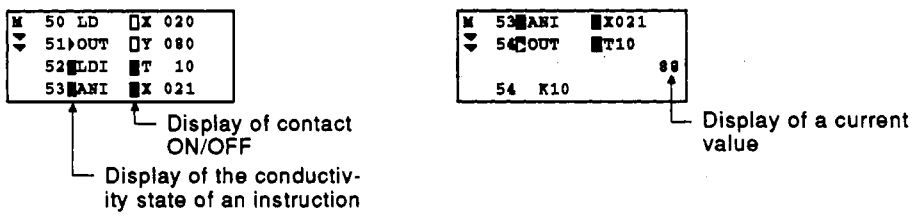
## 6.6 MONITOR (M) Mode Operations

This mode checks operations of ACPU programs. The following checks can be done:

(1) List monitoring

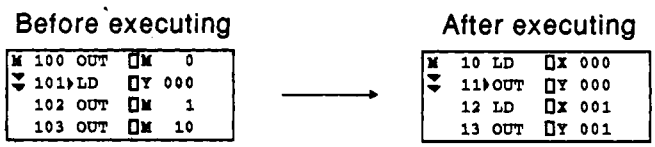
Bit devices can be turned ON/OFF when displaying sequence programs, and the current value of word devices and conductivity status can be monitored.

List monitoring example


M 50 LD X 020
51 OUT Y 080
52 LDI T 10
53 ANI X 021
Display of contact ON/OFF
Display of the conductivity state of an instruction
M 53 ANI X021
54 OUT T10
88
54 K10
Display of a current value

(2) Monitoring search

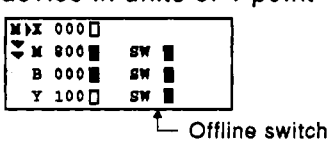
OUT, SET, and RST instructions of a corresponding device can be searched by moving the cursor to the contact instruction. Example of moving the cursor to "LD Y000", searching "OUT Y000", and displaying the corresponding program



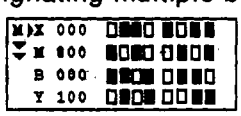

(3) Device monitoring

(a) Monitoring the ON/OFF state and the offline switch state of a bit device

Example of monitoring a bit device in units of 1 point

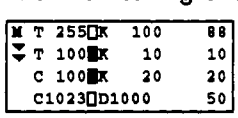



Example of monitoring by designating multiple bit devices



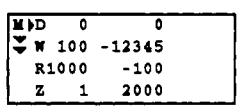

(b) Monitoring the contact of T/C, the set value, and the current value

T/C monitoring example

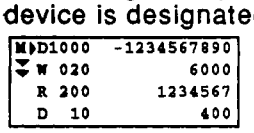



(c) Monitoring the current value of a word device

Word device monitoring example




Monitoring example when a word device is designated to 32 bits

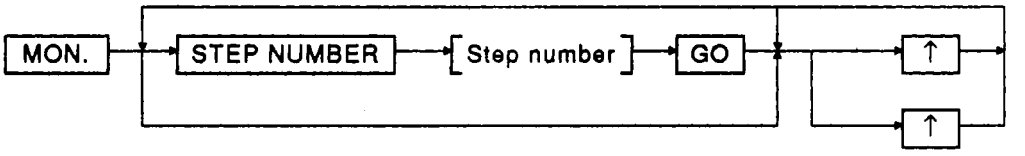


| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | EPROM | |

### 6.6.1 Confirming the operating state by displaying a program (List monitoring)

This operation monitors the ON/OFF state of a contact instruction, the current value of a word device, and conductivity state (whether the instruction is active or not) while displaying a program.

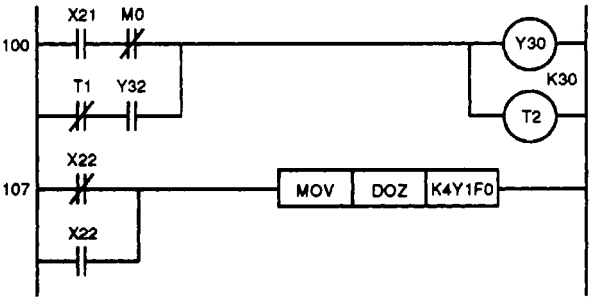
**[Basic operation]**




**[Sample operation]**

A program in the figure below is list monitored

```
      :
100   LD    X21
101   ANI   M0
102   LDI   T1
103   AND   Y32
104   ORB
105   OUT   Y30
106   OUT   T2    K30
107   LDI   X22
108   OR    X23
109   MOV   DOZ   K4Y1F0
      :
```




① MON. → STEP NUMBER → 1 → 0 → 0

```
M   0 LD    X000
 ▼  1 ANI   M100
    2 OR    M101
100
```

Select the MONITOR mode.

② GO

```
M  99 OUT   M200
▼ 100 LD    X021
  101 ANI   M0
  102 LDI   T1
```

Display the program of step 100, and execute list monitoring.

③ GO

```
M 102 LDI   T1
▼ 103 AND   Y032
  104 ORB
  105 OUT   Y030
```

The next screen is displayed.

④ GO

```
M 105 OUT   Y030
▼ 106 OUT   T2
                30
  106  K30
```

The next screen is displayed.

⑤ GO

```
M 106  K30
▼ 107 LDI   X022
  108 OR    X023
  109 MOV
```

The next screen is displayed.

⑥ GO

```
M 109 MOV
▼ 109  DOZ          *
  109  K4Y1F0       *
  114 LD    M100
```

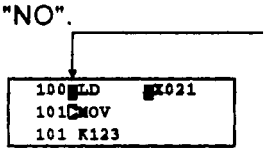
The next screen is displayed.

**[Explanation]**

(1) Not only the program is read by an operation that is the same as the read by the step number, but the conductivity state, ON/OFF state of a device contact, and the current value of a device are also monitored.
However, the conductivity state of an A0J2CPU, A3HCPU, A3MCPU, A2ACPU(S1) and A3ACPU will not be not displayed.

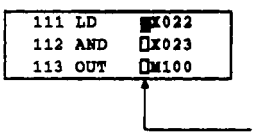
(2) Explanation of display contents

(a) Monitoring conductivity states
Checks whether an instruction such as OUT, MOV, BCD, BIN and operation instruction is executed.
The conductivity state is displayed on the left side of the instruction.
To display the conductivity state, set the conductivity state display setting (see Section 6.9.15) to "YES" in the OTHERS mode. When the power supply is turned ON, the conductivity state display is set to "NO".



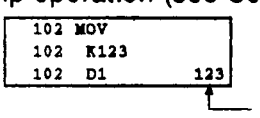

When it is conductive, "■" is displayed; when not conductive, it becomes blank. However, when a conductivity state display and the cursor overlap, "▶" indicates conductivity, and "▸" indicates non-conductivity.

(b) Monitoring bit devices
The ON/OFF state of a bit device is displayed on the left side of the device number.

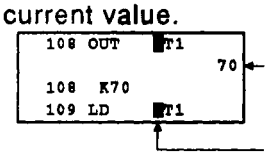



When a bit device turns ON, "■" is displayed, and when a bit device turns OFF "□" is displayed.

(c) Monitoring word devices
The current value of a word device is displayed. A 16-bit instruction displays 1-word data, and a 32-bit instruction displays 2-word data. Change the display format (octal display, decimal display, hexadecimal display and ASCII display) of a current value according to the Help operation (see Section 6.6.4).




(d) T/C monitoring
The contact instruction of T/C monitors the ON/OFF state of a contact.
The OUT T/C instruction monitors the ON/OFF state of a contact and a current value.



(e) Instructions that cannot be monitored
Contacts and current values of instructions (example MOV K4M0 D0) to which the number of digits of a bit device has been designated and instructions (examples LD X000Z and MOV D0 D100Z) modified by indexes cannot be monitored. * is displayed in the monitoring display position.

```
 114 LD     X0Z   *
 116 MOV
 116 K4M0         *
 116 D0      12345
```

(3) After pressing the [STEP NUMBER], step number, and [GO] keys, press the [GO] key to scroll the program up.

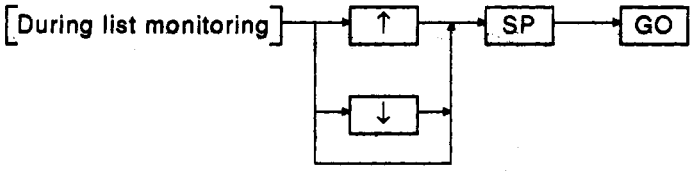
(4) Pressing the [MON.] key in the READ, WRITE, INSERT, or TEST (test of list monitor) modes enables list monitor.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | OPTION | |

### 6.6.2 Searching a coil from the contact (Monitoring search)

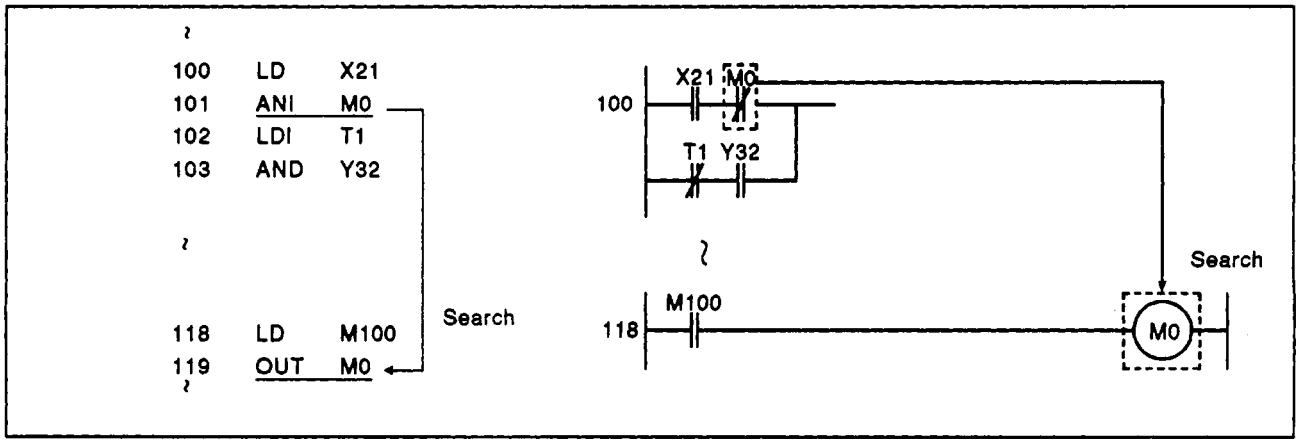
This operation searches and displays OUT, SET, and RST instructions of bit devices (Y, M, L, S, T, C, F, B) used in contact instructions.

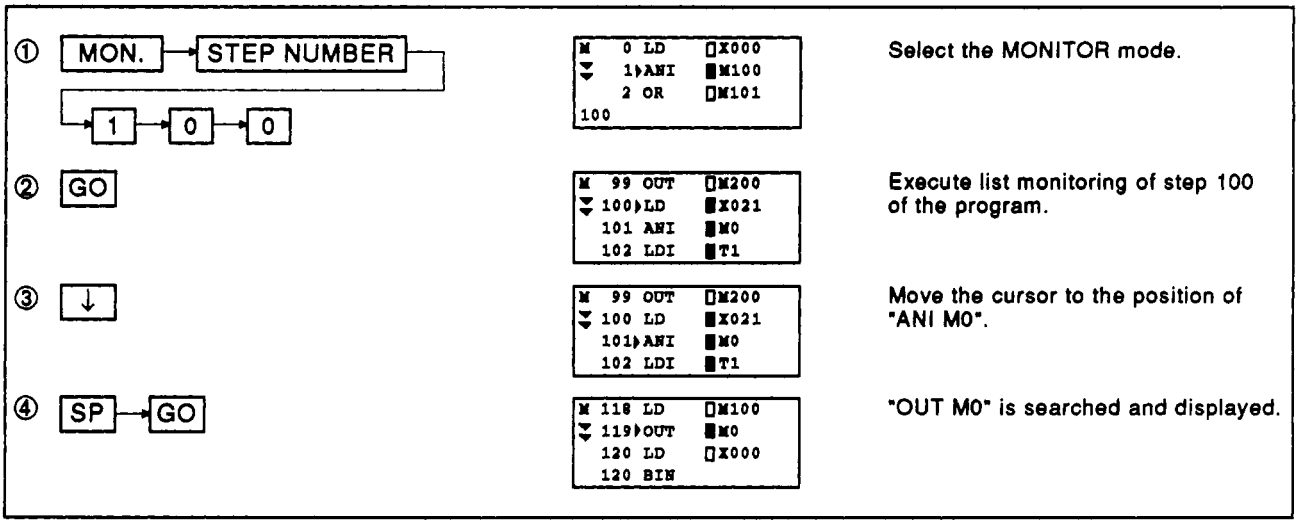
**[Basic operation]**




**[Sample operation]**

The output instruction of M0 is searched from the contact instruction of M0 in the program below



**[Explanation]**

(1) The program of an OUT, SET, and RST instruction of a bit device (Y, M, L, S, T, C, F, B) used in an LD, LDI, AND, ANI, OR, and ORI instruction is read.

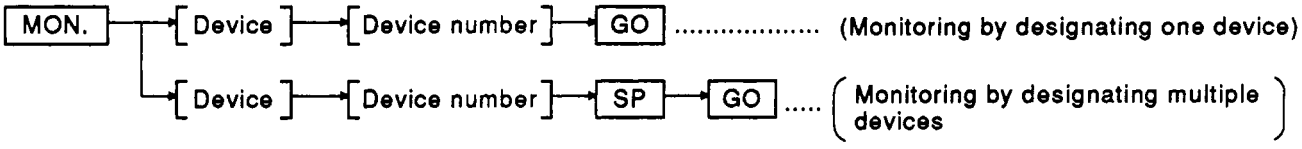
(2) The search is started beginning with step 0 and the program of the step found first is displayed. If there are several OUT, SET and RST instructions, the subsequent programs cannot be searched.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | EPROM | |

### 6.6.3 Device monitoring

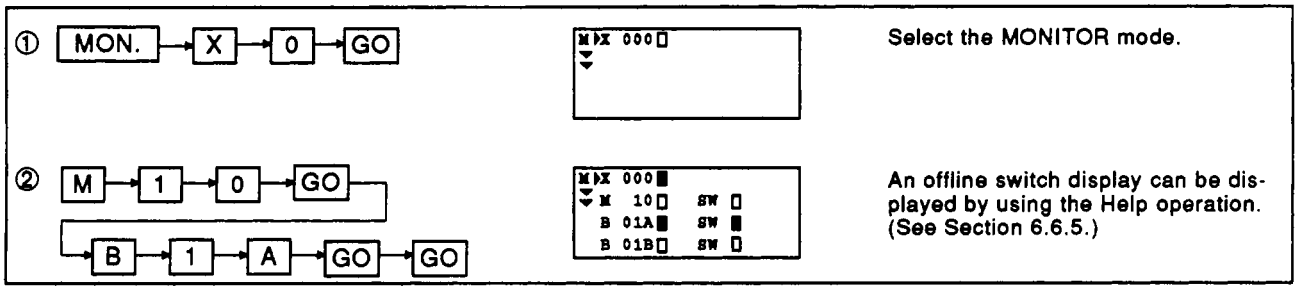
This operation monitors the ON/OFF state and the offline switch state of a bit device and a current value of word device.
1-point designation (monitoring in units of 1 point of a designated device) and multiple-point designation (bit device: monitoring in units of 8 points, word device: monitoring 32-bit data) are used for monitoring.

**[Basic operation]**



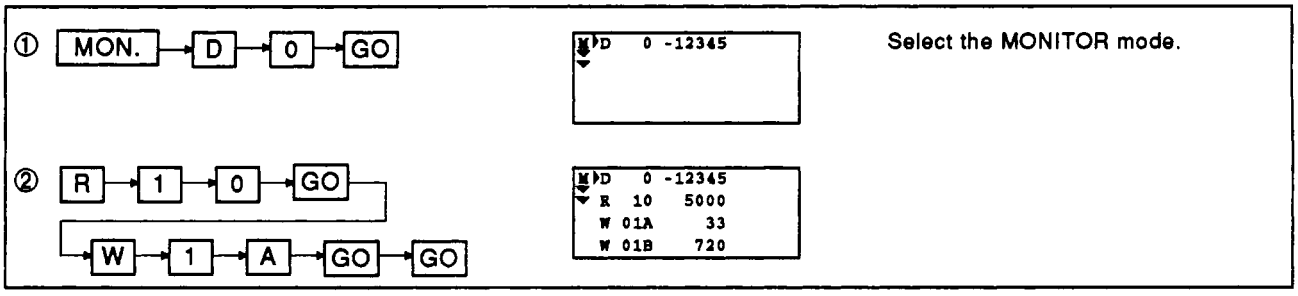

**[Sample operation 1]**

A 1-bit device is designated, and ON/OFF is monitored Monitoring


① MON. X 0 GO
② M 1 0 GO
B 1 A GO GO
M X 000
M X 000
M 10 SW
B 01A SW
B 01B SW
Select the MONITOR mode.
An offline switch display can be displayed by using the Help operation. (See Section 6.6.5.)

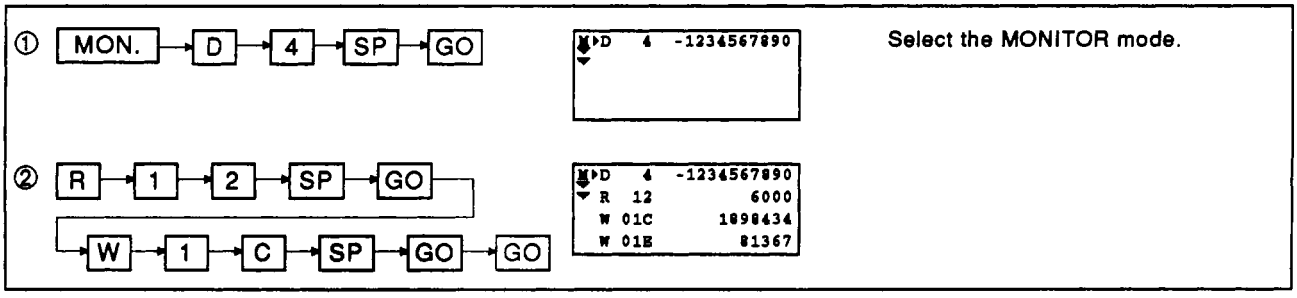
**[Sample operation 2]**

A 1-word device is designated and monitored in 16-bit data Monitoring




**[Sample operation 3]**

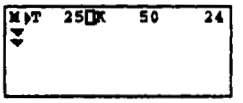
Multiple word devices are designated and monitored in 32-bit data Monitoring



**[Sample operation 4]**

A 1-point timer counter is designated and monitored Monitoring

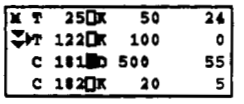
① MON. → T → 2 → 5 → GO



Select the MONITOR mode.

② T → 1 → 2 → 2 → GO

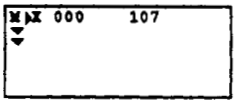
C → 1 → 8 → 1 → GO → GO



**[Sample operation 5]**

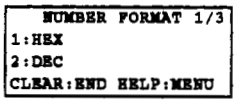
Multiple bit devices are designated and monitored in binary

① MON. → X → 0 → SP → GO



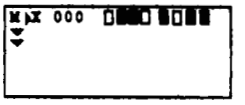
Monitoring in the currently set display format (The screen on the left is displayed when the display format is decimal.).

② HELP → 1




Switch the display format from decimal to binary by using the Help operation.

③ 5



Monitoring multiple bit devices is displayed in binary.

④ M → 1 → 0 → SP → GO

B → 1 → 0 → SP → GO → GO

The device number is processed in units of 8 points. If M10 is set, M8 to M15 are monitored.

**[Explanation]**

(1) Monitoring by designating 1 point

(a) The ON/OFF states of bit devices (X, Y, M, L, S, B) can be monitored. Monitoring of an offline switch state can be also executed. (There is no offline switch function in the A0J2CPU, A2ACPU, A3HCPU, A3MCPU and A3ACPU.)
An offline switch is displayed by using the Help operation or offline switch setting of the TEST Mode. (See Sections 6.6.5 and 6.7.5.)

(b) The current values of word devices (D, W, R) can be monitored.

(c) The set values, current values, and ON/OFF states of contact timer counters (T, C) can be monitored.
However, when the set value is set at a word device, a device number is displayed.

(d) The number of monitored points is 4 max.
However, only 4 points of a bit device, timer/counter, or word device can be monitored.
Monitoring cannot be executed by mixing bit devices, timers, counters, and word devices.

(2) Monitoring by designating multiple points

(a) Designation of multiple bit devices
Monitoring the ON/OFF state of bit devices is executed in units of 8 points.
When a set bit device number cannot be divided by 8, it is monitored automatically by device number in units of 8 points. For example, if multiple M10 are designated, M8 to M15 are monitored.

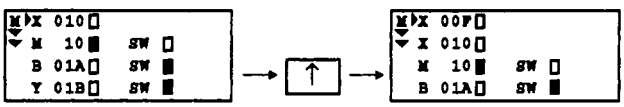
(b) Designation of multiple word devices
These are handled as 32-bit data, and a current values are monitored.

(c) The number of points to be monitored is 4 max.
However, only 4 points of a bit device or word device can be monitored.
Monitoring cannot be executed by mixing bit devices and word devices.

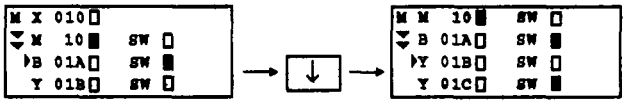
(3) Operation that monitors a serial device number

(a) If the [GO] key is pressed after inputting a device number, the following device number is displayed.

(b) When the cursor is in the top line, a device number immediately before displayed in the top line is monitored by pressing the [ ↑ ] key.



(c) When the cursor is in the third line, the device number next to a device number displayed on the bottom line is monitored by pressing the [ ↓ ] key.




**REMARK**

The current value of a word device can be displayed in binary, octal, decimal, hexadecimal and ASCII code. Change the display format by using the Help operation (see Section 6.6.4).

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | EPROM | |

### 6.6.4 Changing the display format of a current value

This operation switches a current value to binary, octal, decimal, hexadecimal or ASCII display by list monitoring and device monitoring. (Help operation in the MONITOR/TEST/OTHERS modes)

**[Basic operation]**

During operation in the MONITOR/TEST mode → HELP → 1
- 1 ... (Hexadecimal (HEX) display)
- 2 ... (Decimal (DEC) display)
- 3 ... (Octal (OCT) display)
- 4 ... (ASCII (ASCII) display)
- 5 ... (Binary (BIN) display)

During operation by buffer batch monitoring in the OTHERS mode → HELP
- 1 ... (Hexadecimal (HEX) display)
- 2 ... (Decimal (DEC) display)
- 3 ... (Octal (OCT) display)
- 4 ... (ASCII (ASCII) display)

**[Sample operation]**

Changing to hexadecimal display during device monitoring

| | | | |
|---|---|---|---|
| ① | | M▶D 0 -12345<br>▼R 10 5000<br>W 01A 33<br>W 01B 720 | During monitoring in decimal |
| ② | HELP | *** HELP MENU **1/2<br>1:NUMBER FORMAT<br>2:MONITOR<br>CLEAR:END | Press the [HELP] key during monitoring, |
| ③ | 1 | NUMBER FORMAT 1/3<br>1:HEX<br>2:DEC<br>CLEAR:END HELP:MENU | Select "1:NUMBER FORMAT". |
| ④ | 1 | M▶D 0 HCFC7<br>▼R 10 H1388<br>W 01A H0021<br>W 01B H02D0 | Select "1:HEX". The current value of the device is switched to hexadecimal. |

**[Explanation]**

(1) This operation changes the display format of the current position in the MONITOR, TEST, and OTHERS modes.
Set the display format when displaying the MONITOR mode's list monitoring and device monitoring, the TEST mode's word devices (D, R, W) and the OTHERS mode's buffer batch monitoring current values.
Timers/counters (T, C) are displayed only in decimal.

(2) The display format can be binary display, octal display, decimal display, hexadecimal display, or ASCII display.
The display format when the powr supply is turned ON is decimal.
Four kinds of display formats other than binary display are available for buffer batch monitoring in the OTHERS mode.

(3) Some display formats cannot be designated depending on the monitoring device and function. When a display format that cannot be displayed is set, the display format is decimal display.

(a) Binary display
The display format is binary display only when many bit devices are monitored by device monitoring.

(b) Octal display
Devices other than the timer (T)/counter (C) can be displayed in octal.

(c) Decimal display
Monitoring of all devices can be displayed in decimal.

(d) Hexadecimal display
Devices other than the timer (T)/counter (C) can be displayed in hexadecimal.

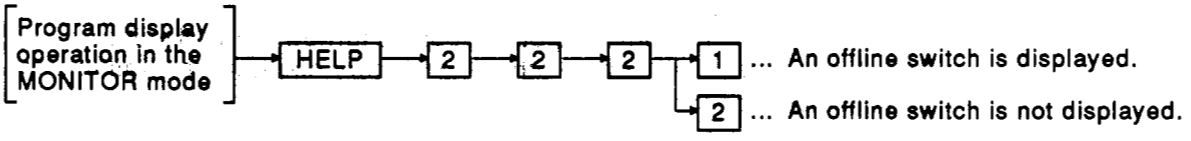
(e) ASCII display
The devices other than the timer (T)/counter (C) can be displayed in ASCII.
Characters can be displayed in the ranges of 20H to 7FH and A0H to DFH. All characters outside these ranges are displayed as dots (..).
(See APPENDIX 4.)

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | EPROM | |

### 6.6.5 Setting the offline switch YES/NO display (Offline switch display)

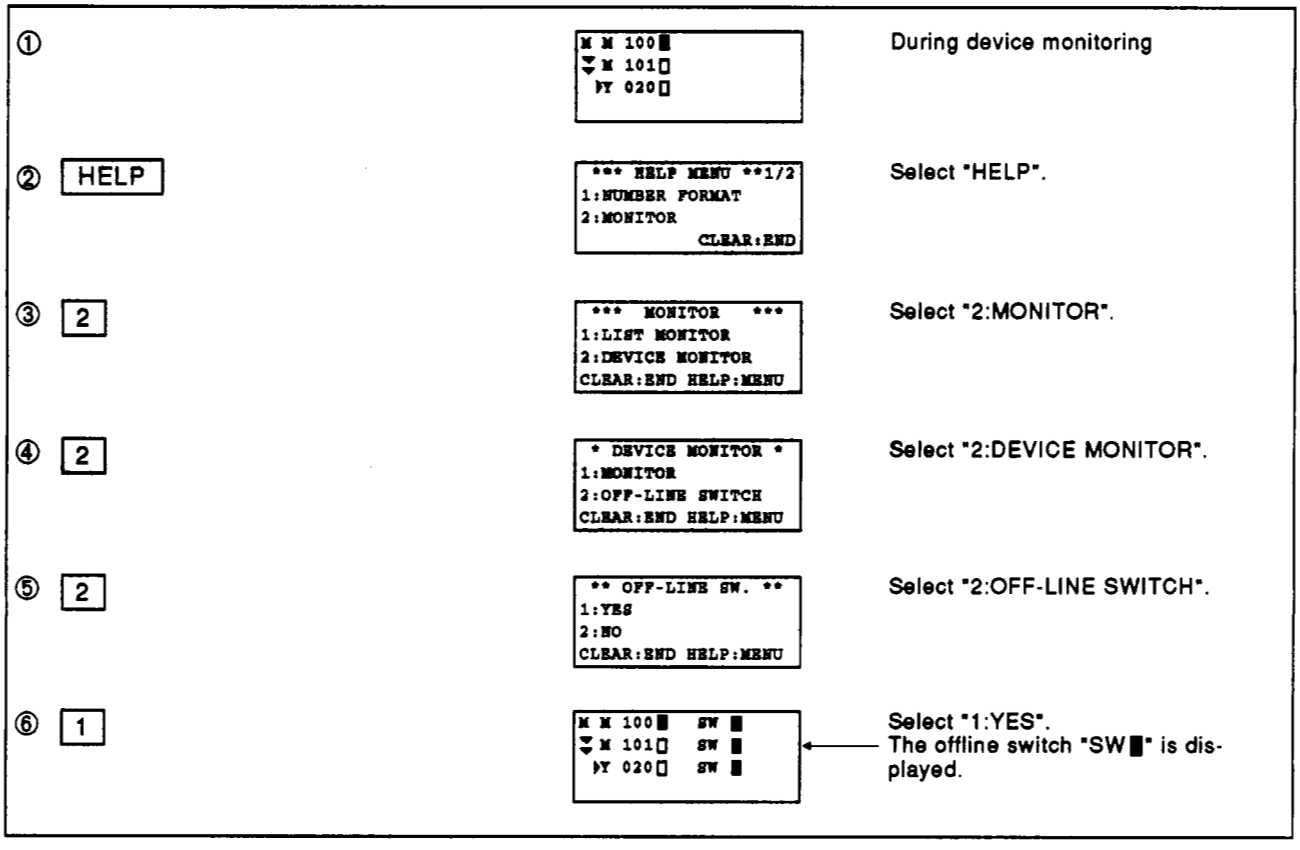
This operation sets the offline switch display YES/NO during device monitoring and device monitoring tests.

**[Basic operation]**




**[Sample operation]**

An offline switch is displayed during device monitoring



**[Explanation]**

(1) When the connected ACPU is an A0J2CPU, A2ACPU, A3HCPU, A3MCPU, or A3ACPU, offline switches are not displayed.

(2) Devices that can display offline switches are Y, M, L, B, and F.

(3) If an offline switch is set or canceled during the device monitoring test in the TEST mode, the offline switch is displayed automatically.

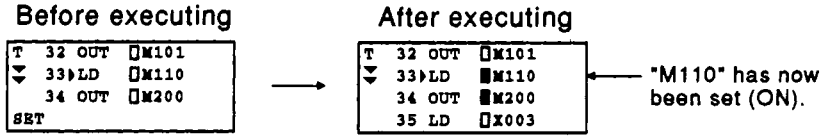
(4) Section 6.7.5 explains how to operate offline switches.

## 6.7 TEST (T) Mode Operations

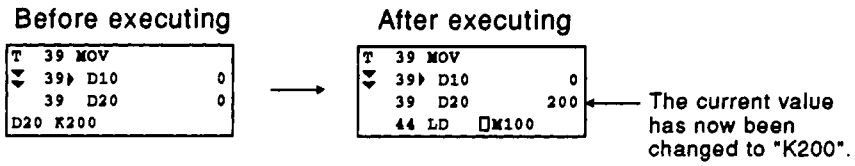
This mode performs test operations of ACPU programs. The following are enabled:

(1) Testing by list monitoring

(a) Setting/resetting bit devices X, Y, M, L, B, and F
Example of setting device "M110" of the cursor position

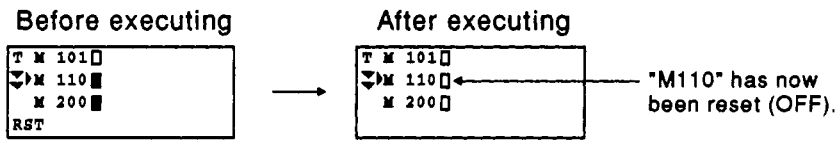



(b) Changing the current value of word devices T, C, D, W, R, A, Z and V (Z1 to Z6 and V1 to V6: only when A2ACPU or A3ACPU is used)
Example of changing the current value of designated device "D20" to "K200"

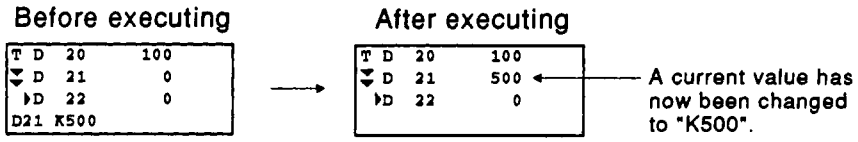



(2) Testing by device monitoring

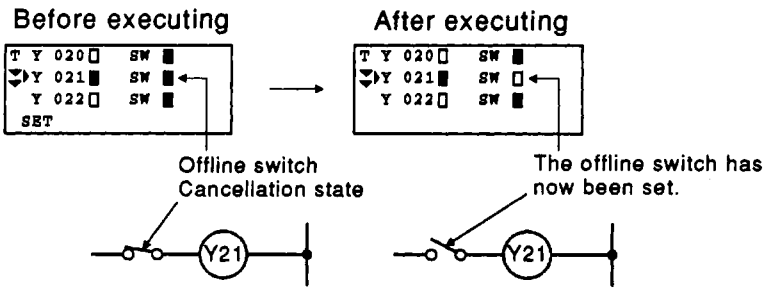
(a) Setting/resetting bit devices X, Y, M, L, B, and F
Example of resetting device "M110" at the cursor position




(b) Changing the current value of word devices T, C, D, W, R, A, Z, and V (Z1 to Z6 and V1 to V6: only when A2ACPU or A3ACPU is used)
Example of changing the current value of designated device "D21" to "K500"




(c) Setting/canceling an offline switch of bit devices Y, M, L, B, and F
Example of setting an offline switch of device "Y21" at the cursor position



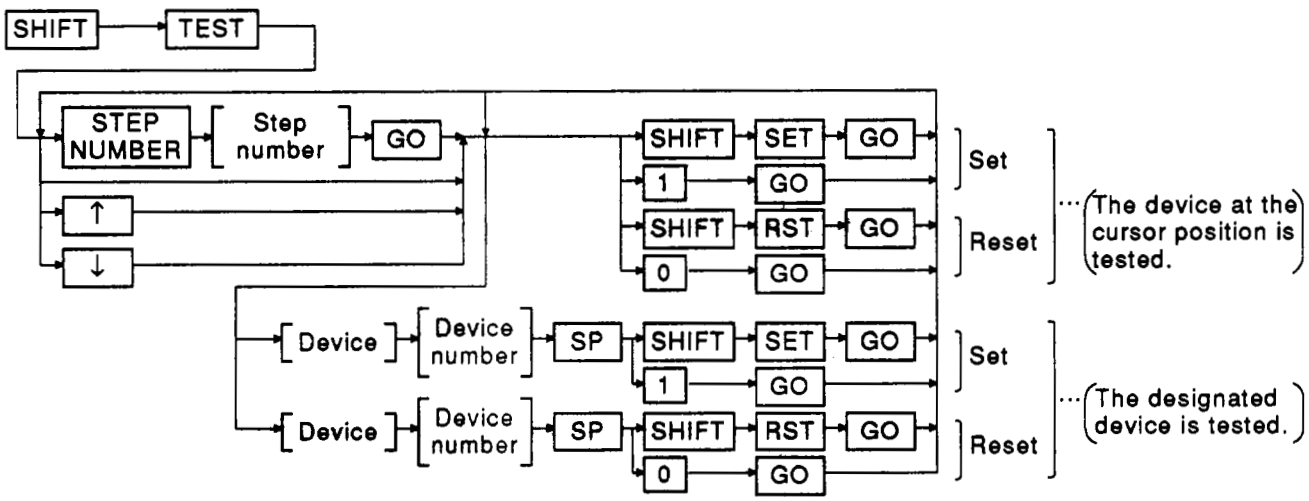
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.7.1 Setting/resetting X, Y, M, L, B, and F by list monitoring (Set/reset of a bit device)

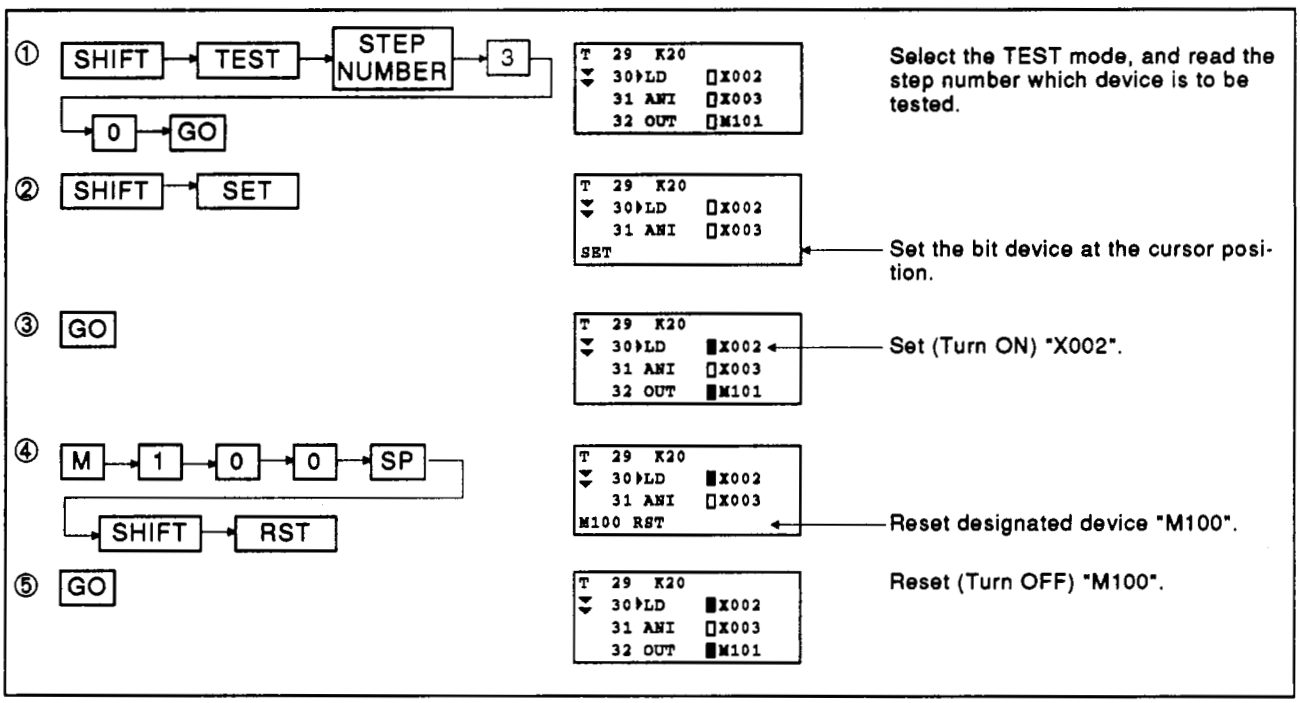
This operation sets/resets (turns ON/OFF) a bit device forcibly by designating a device and a device number.

**[Basic operation]**




**[Sample operation]**

Device "X002" at the cursor position is set and designated device "M100" is reset



**[Explanation]**

(1) Turn ON a device number set by pressing the [SHIFT] → [SET] → [GO] ([1] → [GO]) keys. Turn it OFF by pressing the [SHIFT] → [RST] → [GO] ([0] → [GO]) keys.

(2) As for forcible set/reset operations when the ACPU is in the RUN state, program execution has priority.
Therefore, even if forcible set/reset is executed when an instruction using a correspondent device is being executed, the state only changes for a moment and returns back.

(3) Even if a process input is OFF, when input (X) is set forcibly, an ACPU executes the operation processing as ON.
And, even if input (X) is reset forcibly when a process input is ON, the PC CPU is processed as ON.
If input (X) was set, be sure to reset it after the test operation has been completed.

(4) As for special-function modules, some of the Y numbers cannot be set/reset.

(5) Special relay (M) is controlled by the OS of the CPU. Operate set/reset of special relays after checking the contents utilizing the User's Manual of the connected ACPU.

(6) Pressing the [SHIFT] and [TEST] keys in the READ, WRITE, INSERT, DELETE, and MONITOR modes (list monitoring) enables the test operation of the designated device indicated by the cursor ( ▶ ).

**IMPORTANT**

Be sure to reset devices set by this operation before operating.
Batch reset can be executed by using the CPU's reset switch.

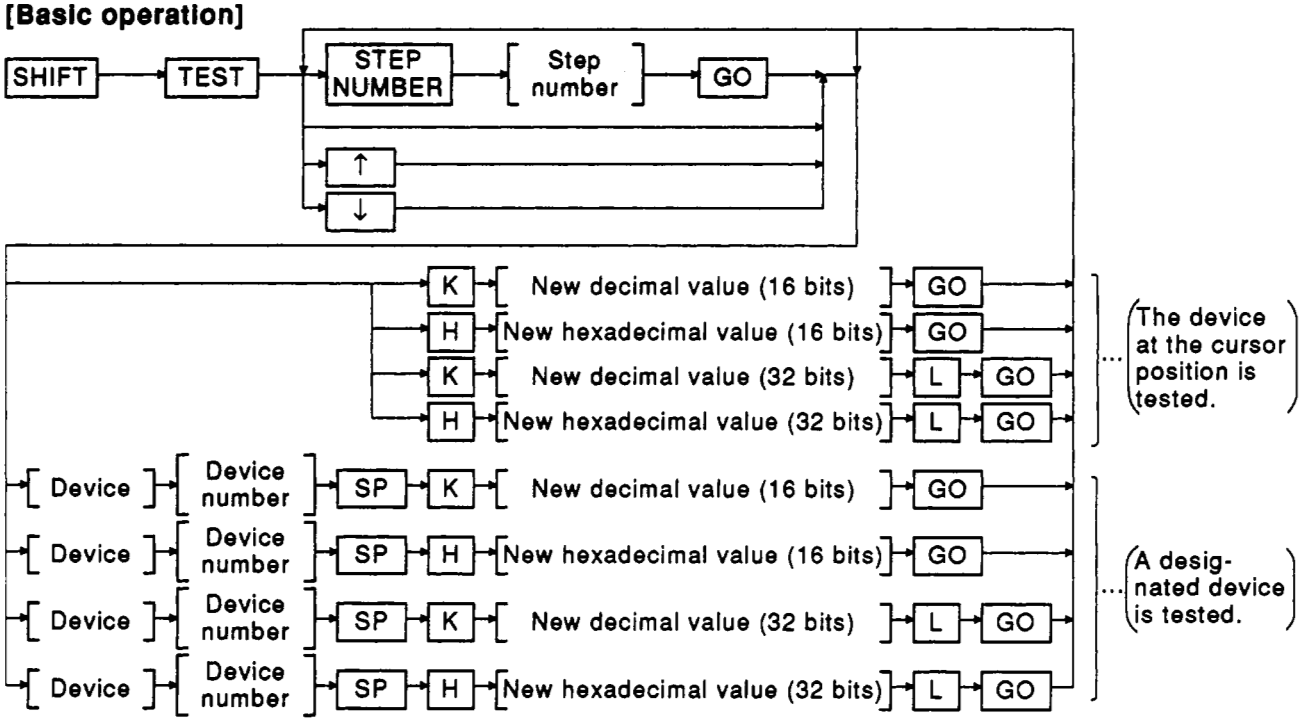
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.7.2 Changing the current values of T, C, D, W, R, A, Z, or V by list monitoring (Changing current values of word devices)

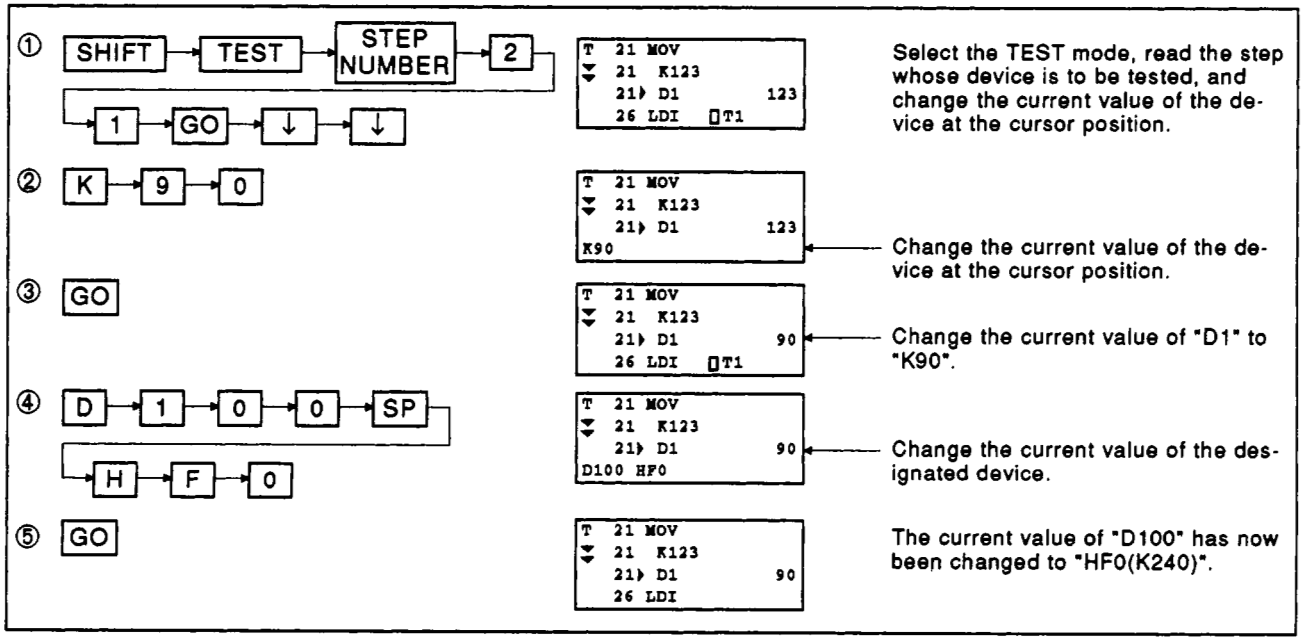
This operation changes the current value of a word device forcibly by designating a device and a device number.

**[Basic operation]**




**[Sample operation]**

The current value of device "D1" at the cursor position is changed to "K90", and the current value of designated device "D100" is changed to "HF0(K240)"



**[Explanation]**

(1) The current value of a device changes according to the contents of the sequence program being processed when the ACPU is in the RUN state.

(2) Special registers are managed by the OS of ACPU.
When changing the current value of a special register, operate after confirming the contents by utilizing the User's Manual of the connected ACPU.

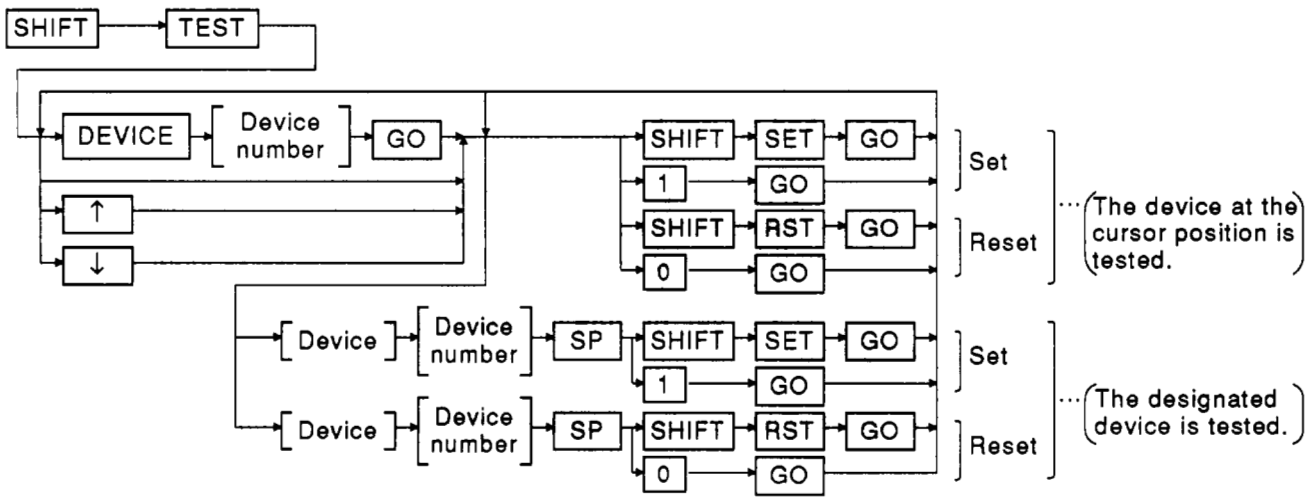
(3) Pressing the [SHIFT] and [TEST] keys in the READ, WRITE, INSERT, DELETE, and MONITOR modes (list monitoring) enables the test operation of the designated device indicated by the cursor ( ▶ ).

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | EPROM | |

### 6.7.3 Setting/resetting X, Y, M, L, B, or F by device monitoring (Bit device set/reset)

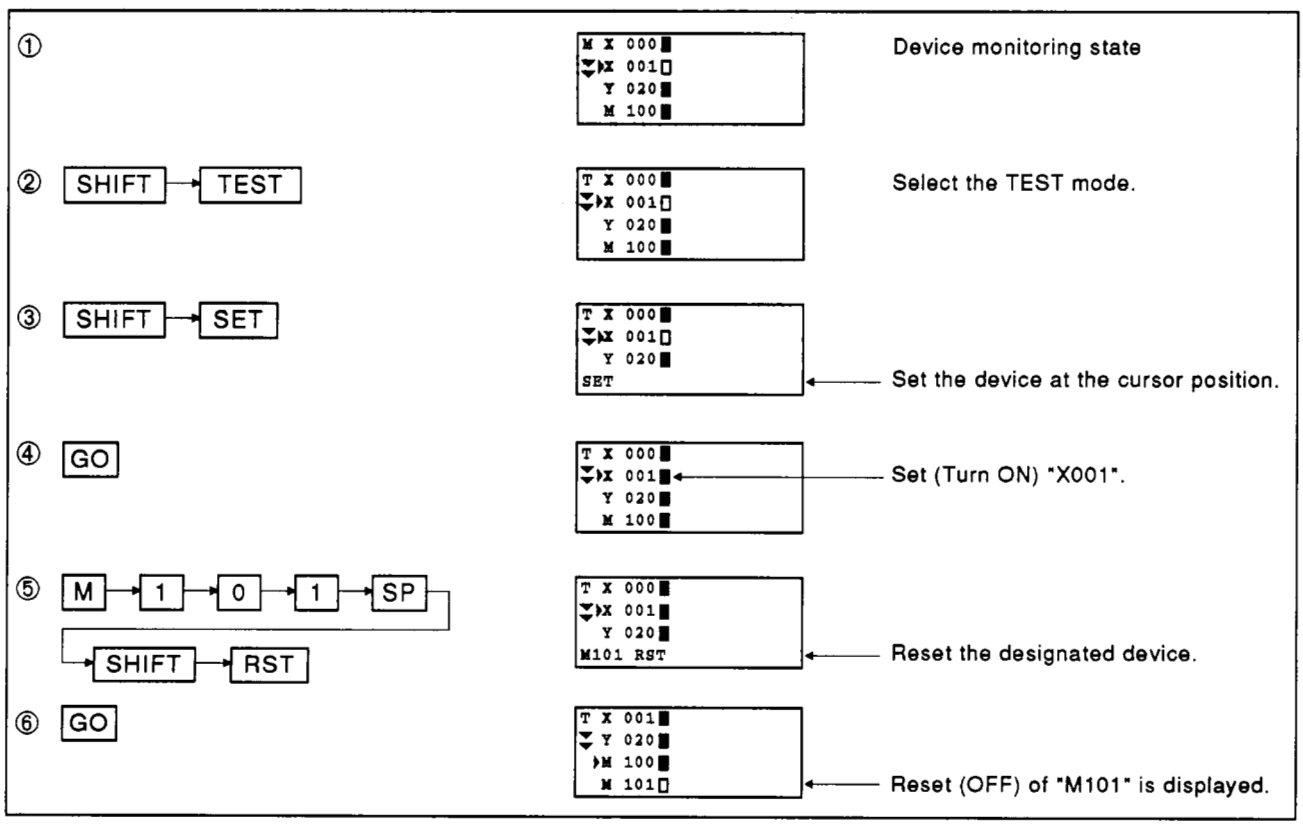
This operation sets/resets (turns ON/OFF) a bit device forcibly by designating a device and a device number.

**[Basic operation]**




**[Sample operation]**

Device "X001" at the cursor position is set and designated device "M101" is reset



**[Explanation]**

(1) Turn ON a device number set by pressing the [SHIFT] → [SET] → [GO] ([1] → [GO]) keys.
Turn it OFF by pressing the [SHIFT] → [RST] → [GO] ([0] → [GO]) keys.

(2) When a device that is not displayed is set or reset, the corresponding device and device state are displayed.

(3) As for forcible set/reset operations when the ACPU is in the RUN state, program execution has priority.
Therefore, even if forcible set/reset is executed when an instruction using a correspondent device is being executed, the state only changes for a moment and returns back.

(4) Even if a process input is OFF, when input (X) is set forcibly, an ACPU executes the operation processing as ON.
And, even if input (X) is reset forcibly when a process input is ON, the PC CPU is processed as ON.
If input (X) was set, be sure to reset it after the test operation has been completed.

(5) As for special-function modules, some of the Y numbers cannot be set/reset.

(6) Special relay (M) is controlled by the OS of the CPU. Operate set/reset of special relays after checking the contents utilizing the User's Manual of the connected ACPU.

(7) Pressing the [SHIFT] and [TEST] keys in the MONITOR mode (list monitoring) enables the test operation of the designated device indicated by the cursor ( ▶ ).
When the [SHIFT] and [TEST] keys are pressed in the READ, WRITE, INSERT, DELETE, or MONITOR (list monitoring) mode, the test operation is enabled by inputting the designated device and device number.

**IMPORTANT**

Be sure to reset devices set by this operation before operating.
Batch reset can be executed by using the CPU's reset switch.

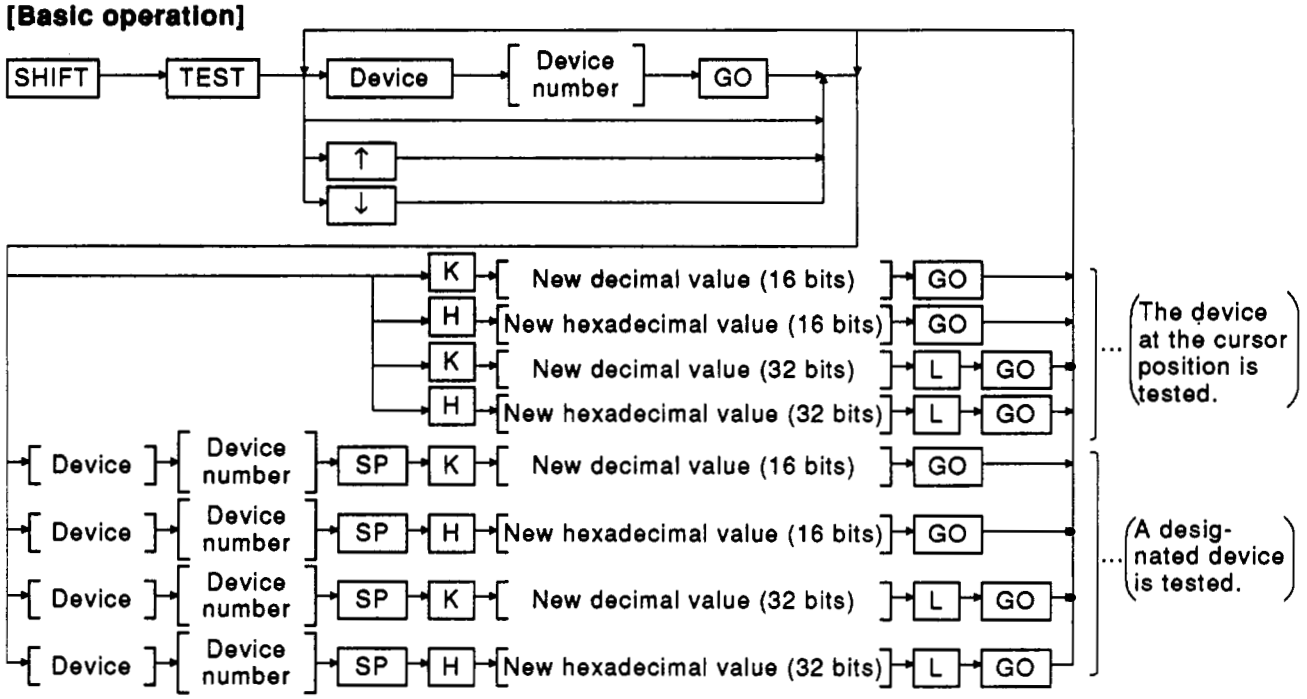
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.7.4 Changing the current value of T, C, D, W, R, A, Z, or V by device monitoring (Current value change of a word device)

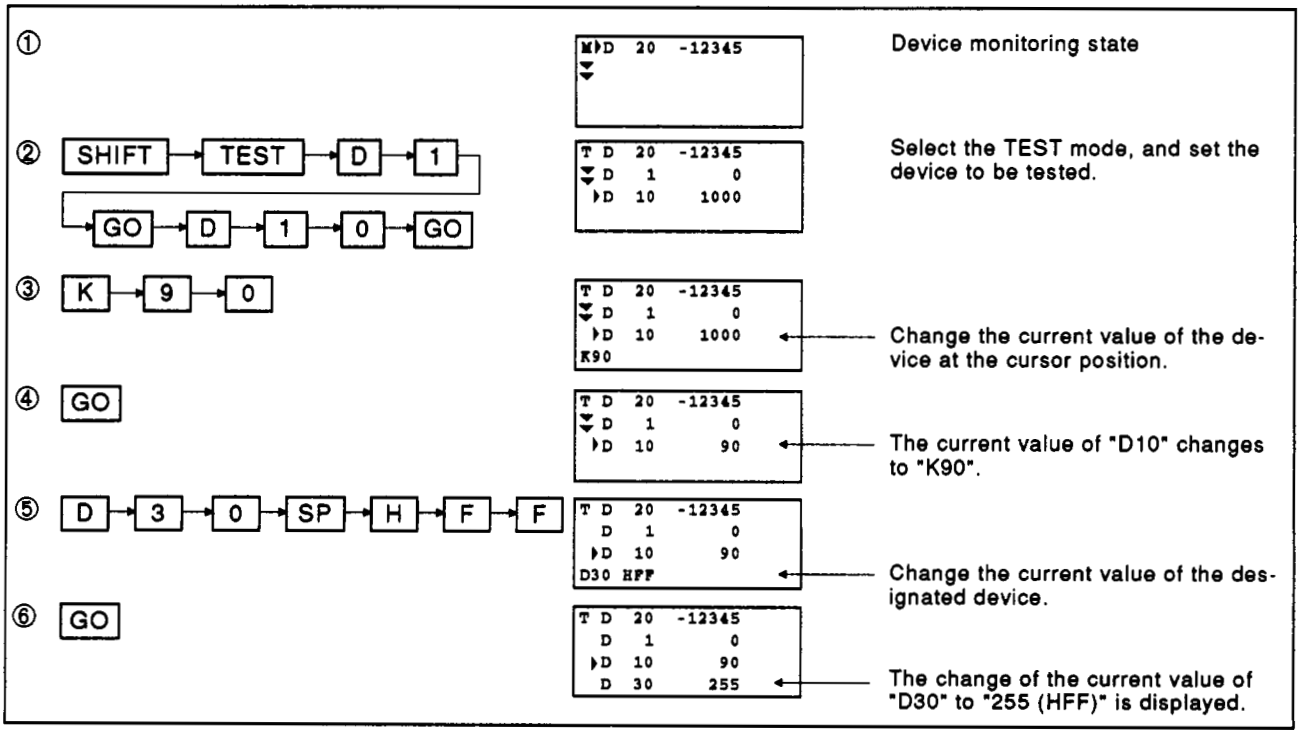
This operation changes the current value of a word device forcibly by designating a device and a device number.

**[Basic operation]**




**[Sample operation]**

The current value of device "D10" at the cursor position is changed to "K90", and the current value of designated device "D30" is changed to "HFF (K255)"



**[Explanation]**

(1) When a device that is not displayed is set or reset, the corresponding device and device state are displayed.

(2) The current value of a device changes according to the contents of the sequence program being processed when the ACPU is in the RUN state.

(3) Special registers are managed by the OS of ACPU.
When changing the current value of a special register, operate after confirming the contents by utilizing the User's Manual of the connected ACPU.

(4) Pressing the [SHIFT] and [TEST] keys in the MONITOR mode (list monitoring) enables the test operation of the designated device indicated by the cursor ( ▶ ).
When the [SHIFT] and [TEST] keys are pressed in the READ, WRITE, INSERT, DELETE, or MONITOR (list monitoring) mode, the test operation is enabled by inputting the designated device and device number.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | EPROM | |

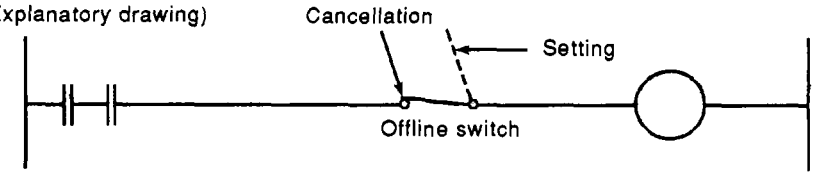
### 6.7.5 Setting/canceling an offline switch of Y, M, L, B, or F by device monitoring (Setting/cancellation of an offline switch)

This operation sets and cancels the offline switch that separates the device (Y,M,L,B, and F) used in an OUT instruction from the operation processing of a sequence program.
The device of the device number that set an offline switch is separated from the operation processing of a sequence program. It can be turned ON and OFF optionally by set/reset operation of the bit device (see Sections 6.7.1 and 6.7.3).

Setting of an offline switch ..... Separated from the operation processing of a sequence program.

Canceling of an offline switch ... It is processed according to the operation of the sequence program.

(Explanatory drawing)



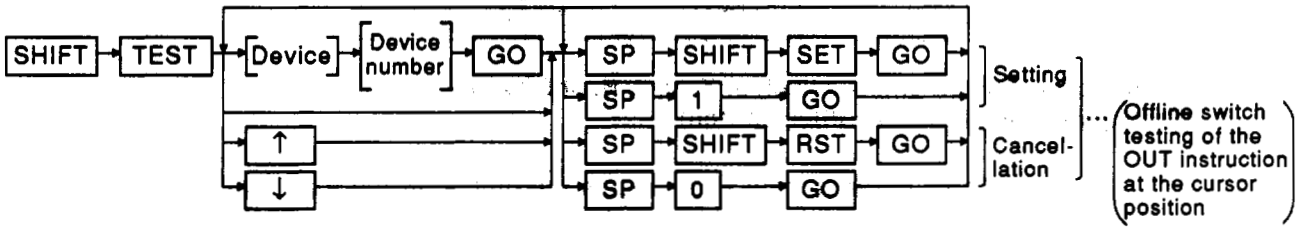

If the connected ACPU is an A0J2CPU, A2ACPU, A3HCPU, A3MCPU or A3ACPU, this operation cannot be executed.

**POINTS**

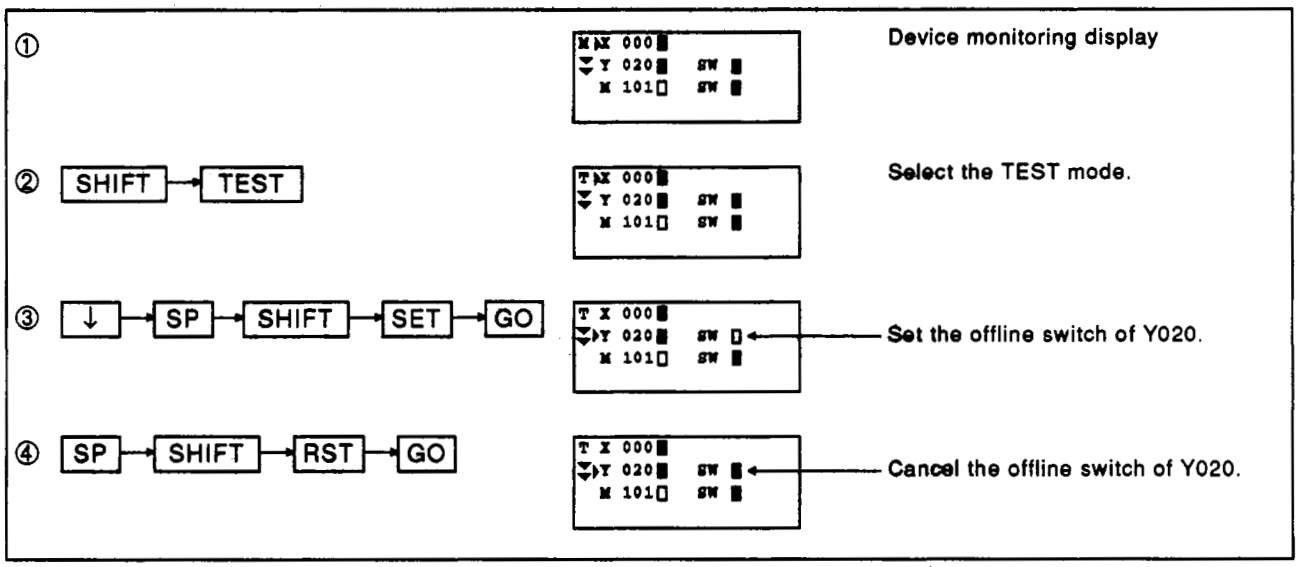
(1) An offline switch is displayed automatically if the offline switch is set or canceled in the state for not displaying an offline switch.
An offline switch is displayed by the "Offline SW setting" of the [HELP] operation in the MONITOR mode.

(2) An offline switch is displayed only during device monitoring in the MONITOR and TEST modes.

**[Basic operation]**




**[Sample operation]**




**[Explanation]**

(1) Be sure to set the cursor ( ▶ ) at the device which sets/cancels the offline switch.
The set device number goes into the offline state by pressing the [SP] → [SHIFT] → [SET] → [GO] ([SP] → [1] → [GO]) keys, and it goes into the online state by pressing the [SP] → [SHIFT] → [RST] → [GO] ([SP] → [0] → [GO]) keys.

(2) As for special-function module, some Y numbers cannot set offline switches.

(3) Pressing the [SHIFT] and [TEST] keys in the MONITOR mode (device monitoring) enables the offline switch operation of the designated device indicated by the cursor ( ▶ ).
When the [SHIFT] and [TEST] keys are pressed in the READ, WRITE, INSERT, or MONITOR (list monitoring) mode, the offline switch operation is enabled by inputting the designated device and device number.

## 6.8 Parameter Setting

This section explains how to clear all parameters of an ACPU and explains how to set the program capacity, file register capacity, etc.
Only those parameters that can be set by the A8PUE are given on the parameter sheet attached to the APPENDIX. An A6GPP/A6PHP (SW[ ]GP-GPPA), IBM* PC/AT (SW[ ]IX-GPPAE) and A7PHPE (SW[ ]RX-GPPAE), etc. are necessary for setting of other parameters.
Since this manual only explains parameter setting operations, determine the setting contents of parameters by referring to the Programming Manual of the particular ACPU.
The operation given below illustrates parameter setting.

* IBM is a registered trade mark of the International Business Machines Corporation.

PARAMETER mode

- Parameter all clear ..... The contents of a parameter in the A8PUE are set to default.
- Parameter setting ..... Parameters in the A8PUE are set.

  In the case of ACPUs other than an A0J2CPU

  - 1. Memory capacity setting... Program capacity and file register capacity are set.
  - 2. M, L, S setting ..... The internal relay (M), latch relay (L), and step relay (S) ranges are set.
  - 3. Timer setting ..... The low-speed, high-speed, and addition timer ranges are set.
  - 4. Counter function ..... The number of device points of an extension counter and the set value head device are set.
  - 5. Latch range setting..... The latch ranges of B, T, C, D, and W are set.
  - 6. WDT setting..... WDT is set.
  - 7. I/O control ..... The I/O control method is set.
  - 8. Key word setting ..... Key words of 6 digits or less are registered in hexadecimal.

  In the case of an A0J2CPU

  - 1. Latch range setting..... The latch ranges of M (L, S), B, T, C, D, and W are set.
  - 2. Step relay setting ..... The range of step relay (S) is set.

- Parameter writing ..... A parameter in the A8PUE is written to the ACPU.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | EPROM | |

### 6.8.1 All clearing of parameters

This operation clears all parameters written to the ACPU and sets them to default.

**[Basic operation]**

PARAM. → 1 → ↑ → GO → END → ↑ → GO

**[Sample operation]**

All the parameters of the ACPU are cleared and are set to default

| | | | |
|---|---|---|---|
| ① | PARAM. | P▶1 PARAM. ALL CLEAR<br>2 PARAM. SETTING<br><END> TO WRITE | Select the PARAMETER mode. |
| ② | 1 | P PARAM. ALL CLEAR<br>YES<br>▶ NO | Select "1:PARAM. ALL CLEAR". |
| ③ | ↑ | P PARAM. ALL CLEAR<br>▶ YES<br>NO | Move the cursor to "YES". |
| ④ | GO | P▶1 PARAM. ALL CLEAR<br>2 PARAM. SETTING<br><END> TO WRITE | When the parameters in the A8PUE are set to default, the screen shown on the left is displayed. |
| ⑤ | END | P PARAM. SETTING<br>YES<br>▶ NO | Select YES. |
| ⑥ | ↑ → GO | P PARAM. SETTING<br>▶ YES<br>NO<br>COMPLETED | When the writing of parameters in the A8PUE to an ACPU have been completed, "COMPLETED" is displayed.<br>(Pressing the [GO] key when the cursor is at "NO" returns to the ④ operation.) |

**[Explanation]**

(1) Operation of "1:PARAM. ALL CLEAR" sets the parameters in the A8PUE to default.

(2) If writing to ACPU has not been executed and the mode is transferred to the OTHERS mode after operating "1: PARAM. ALL CLEAR", the parameters in the A8PUE return to the setting before operating "1: PARAM. ALL CLEAR".

(3) If the key word is registered, parameter all clear cannot be executed.

**REMARK**

The parameter in the A8PUE is set to a set value read from ACPU when starting up the A8PUE.

**POINTS**

- When executing parameter all clear, if the key word registered in the ACPU is unknown, see the "PC memory all clear" operation shown in Section 5.1.3. And then, clear (delete) the unknown registered key word.
  If "PC memory all clear" is executed, all other user data (such as sequence programs) will also be cleared.
- Section 6.8.2 gives details about when (a) the key word registered in an ACPU is changed, or (b) a new key word is registered.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

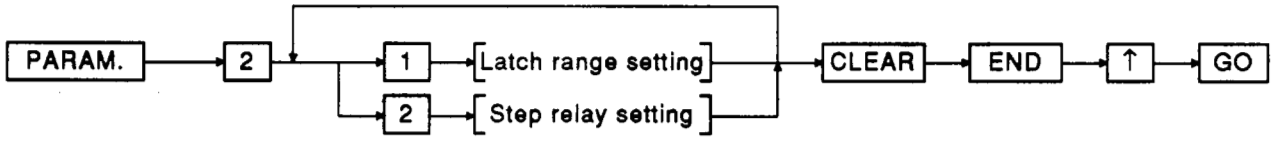
| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.8.2 Parameter setting

This operation sets parameters and writes them to an ACPU.
The operation of each type of ACPU type (other than an A0J2CPU and A0J2CPU) is explained.

**1. In the case of an A0J2CPU**

**[Basic operation 1]**




**[Sample operation 1]**

A0J2CPU parameters have now been set

| | | | |
|---|---|---|---|
| ① | PARAM. | P▶1 PARAM. ALL CLEAR<br>2 PARAM. SETTING<br><END> TO WRITE | Select the PARAMETER mode. |
| ② | 2 | P▶1 LATCH RANGE SET<br>2 S RELAY SETTING | Select "2:PARAM. SETTING". |
| ③ | 1 → ↑ | P LATCH RANGE SET<br>▶ NO LATCH<br>1/2 LATCH<br>ALL LATCH | Sample operation for selecting "1:LATCH RANGE SET" and setting "NO LATCH" |
| ④ | GO → 2 → ↓ | S RELAY SETTING<br>NONE<br>▶PART 1536-2047 | Sample operation for selecting "2:S RELAY SETTING" and setting "PART 1536-2047" |
| ⑤ | GO → CLEAR | P 1 PARAM. ALL CLEAR<br>▶2 PARAM. SETTING<br><END> TO WRITE | Display of the parameters menu |
| ⑥ | END | P PARAM. SETTING<br>YES<br>▶ NO | Select NO. |
| ⑦ | ↑ → GO | P PARAM. SETTING<br>▶ YES<br>NO<br>COMPLETED | When the writing of parameters in the A8PUE to an ACPU has been completed, "COMPLETED" is displayed.<br>(Pressing the [GO] key when the cursor is at "NO" returns to the ⑤ operation.) |

**[Explanation 1]**

(1) Latch range setting
Select the latch range from not latched, 1/2 latch, or all latched.

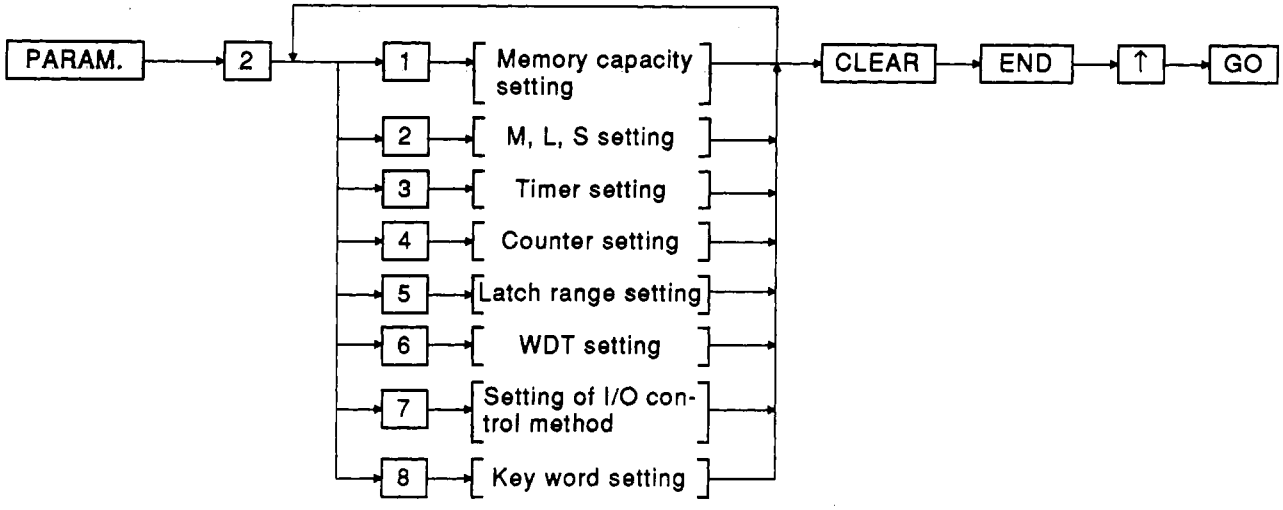
| Latch Items | Latch Ranges |
|---|---|
| Not latched | ——— |
| 1/2 latch | L1024 to 2047 (L1024 to 1535, S1536 to 2047)<br>T40 to 79/T100 to 119/T124 to 127<br>C64 to 127<br>D256 to 511<br>B200 to 3FF (W200 to 3FF) |
| All latched | L0 to 2047 (L0 to 1535, S1536 to 2047)<br>T0 to 127<br>C0 to 127<br>D0 to 511<br>B000 to 3FF (W000 to 3FF) |

(2) Step relay setting
PART (S1536 to 2047)/NONE of the step relay is set.

(3) If writing to ACPU has not been executed and the mode is transferred to the OTHERS mode after operating "1: PARAM. SETTING", the parameters in the A8PUE return to the setting before operating "2: PARAM. SETTING".

### 2. In the case of ACPUs other than an A0J2CPU

**[Basic operation 2]**




**[Sample operation 2]**

Operation from "1. MEM.CAPACITY SET" until "8. KEY WORD" is set (Operation of the previous parameter item is shown in this example. However, some items cannot be set depending on the type of ACPU model to which the A8PUE is connected.)

| | | | |
|---|---|---|---|
| ① | PARAM. | P▶1 PARAM. ALL CLEAR<br>2 PARAM. SETTING<br><br><END> TO WRITE | Select the PARAMETER mode. |
| ② | 2 | P▶1 MEM.CAPACITY SET<br>2 M,L,S SETTING<br>3 TIMER SETTING<br>↓ 4 COUNTER SETTING | Select "2:PARAM. SETTING".<br>(When the operation of each item has been completed, the screen returns to this display.) |
| ③ | 1 → 4 → GO → 2 | P MEM.CAPACITY SET<br>PROGRAM 1- [ ]KS<br>F REG. 0- [ ]KP | Set the program capacity to 4 Ksteps, and set the file register to 2 Kpoints. |
| ④ | GO | P▶1 MEM.CAPACITY SET<br>2 M,L,S SETTING<br>3 TIMER SETTING<br>↓ 4 COUNTER SETTING | Complete "1:MEM. CAPACITY SET". |
| ⑤ | 2 → CLEAR → GO | P M,L,S SETTING<br>M[ 0-] L[ -]<br>S[ -] | Set the latch relay to 0 point. |
| ⑥ | GO | P 1 MEM.CAPACITY SET<br>▶2 M,L,S SETTING<br>3 TIMER SETTING<br>↓ 4 COUNTER SETTING | Complete "2:M, L, S SETTING". |
| ⑦ | 3 → 5 → 1 → 2 → GO<br>D → 1 → 0 → 2 → 4<br>GO | P TIMER SETTING<br>EXT.T 0-2048 [ 512]<br>SET D,R,W [D1024]<br>↓ | Set the number of timer device points to 512 points in the A3ACPU, and set the head device number in which the set value is stored to D1024. |

⑧ GO → CLEAR → GO → GO

```
P   TIMER SETTING
  T(100ms)[    0-]
  T(10ms )[     -]
↓ T(RET. )[     -]
```

Set T0 to T255 to the 100 msec timer .

⑨ GO → 2 → 5 → 6 → GO

```
P   TIMER SETTING
↑ T(100ms)[     -]
  T(10ms )[ 256-]
  T(RET. )[     -]
```

Set T256 to T511 to 10 the msec timer.

⑩ GO

```
P 1 MEM.CAPACITY SET
  2 M,L,S SETTING
 ▶3 TIMER SETTING
↓ 4 COUNTER SETTING
```

Complete "3:TIMER SETTING".

⑪ 4 → 4 → 8 → 0 → GO
D → 1 → 2 → 8 → 0

```
P   COUNTER SETTING
EXT.T 0-1024   [ 480]
SET D,R,W      [D1280]
```

Set the number of counter device points to 480 points in A3ACPU, and set the head device number in which the set value is stored to D1280.

⑫ GO

```
P 1 MEM.CAPACITY SET
  2 M,L,S SETTING
  3 TIMER SETTING
↓▶4 COUNTER SETTING
```

Complete "4:COUNTER SETTING".

⑬ 5 → ↓ → ↓ → ↓ → ↓ → ↓
5 → 1 → 2 → GO → 1
0 → 2 → 3

```
P   LATCH RANGE SET
↑ C       [     -     ]
  D       [ 512-1023]
↓ RANGE    0-1023
```

Sample operation of setting D512 to D1023 to the latch in the A3ACPU

⑭ GO → ↓ → ↓ → ↓ → ↓
GO → GO

```
P▶5 LATCH RANGE SET
  6 WDT SETTING
  7 I/O CONTROL
  8 KEY WORD
```

Complete "5:LATCH RANGE SET".

⑮ 6 → 1 → 5 → 0

```
P   WDT SETTING
          [ 150]ms
  RANGE   10-2000
```

Set the WDT (watchdog timer) setting to 150 msec.

⑯ GO

```
P 5 LATCH RANGE SET
 ▶6 WDT SETTING
  7 I/O CONTROL
  8 KEY WORD
```

Complete "6:WDT SETTING".

⑰ 7 → ↑

```
P   I/O CONTROL
   DIRECT [    . Y ]
   REFRESH [ X .   ]
USE CURSOR TO SELECT
```

Set the I/O control method to input:REFRESH and output:DIRECT.

⑱ GO

```
P 5 LATCH RANGE SET
  6 WDT SETTING
 ▶7 I/O CONTROL
  8 KEY WORD
```

Complete "7:I/O CONTROL".

⑲ 8 → 8 → 6 → 0 → 2
0 → 2

```
P   KEY WORD

    [860202]
```

Set the key word to "860202".

⑳ GO

```
P 5 LATCH RANGE SET
  6 WDT SETTING
  7 I/O CONTROL
 ▶8 KEY WORD
```

Complete "8:KEY WORD".

㉑ CLEAR

```
P 1 PARAM. ALL CLEAR
 ▶2 PARAM. SETTING

<END> TO WRITE
```

Display of the menu of a parameter

| | | | |
|---|---|---|---|
| ㉒ | END | P PARAM. SETTING<br>YES<br>▸ NO | Select YES. |
| ㉓ | ↑ → GO | P PARAM. SETTING<br>▸ YES<br>NO<br>COMPLETED | When the writing of a parameter in the A8PUE to an ACPU has been completed, "COMPLETED" is displayed.<br>Pressing the [GO] key when the cursor is at "NO" returns to the ㉑ operation. |

**[Explanation 2]**

(1) Memory capacity setting
Set the main sequence program capacity and the file register capacity.
When an A3(N), A3H A3M, A3V, A73, A373, or A3ACPU is used, the setting of a subsequence program capacity is enabled.

(2) M, L, S setting

(a) When an A0J2H, A2C, A1(N), A2(N), A3(N), A3H, A3M, A3V, A73, or A373CPU is used, set the head device number that is used by latch relay/step relay.

(b) When an A2A or A3ACPU is used, set the head device number that is used by latch relay/step relay/internal relay.

(c) Erase the display of the head device number of a device not to be used by pressing the [CLEAR] key. It can be set to 0 points.

(3) Timer setting

(a) When an A0J2H, A2C, A1(N), A3(N), A3H, A3M, A3V, A73, or A373CPU is used, set the head device number that is used for low-speed/high-speed/addition timer.

(b) When an A2A or A3ACPU is used
Set the number of timer device points, the head device number in which a set value after T256 is stored and the head device number to be used by low-speed/high-speed/addition timer.
Set the head device number that is used for the range of T0 to 255 and the range after T256 by low-speed/high-speed/addition timer when the number of timer device points exceeds 256 points.

(c) Erase the display of the head device number of a device not to be used by pressing the [CLEAR] key. It can be set to 0 points.

(4) Counter setting
Set the head device number in which the number of counter device points and the set value after C255 are stored.
This setting is necessary only when using an A2A or A3ACPU.

(5) Latch range setting

(a) Set the range of a device to be set to latch.
The allowable setting range is displayed on the bottom line of the screen.

(b) Erase the display of the head device number of a device not to be used by pressing the [CLEAR] key. It can be set to 0 points.

(6) WDT setting
Set the set value of WDT to 10 msec units.

(7) Setting of the I/O control method
When an A3H or A3MCPU is used, set an I/O control method.

(8) Setting of a key word

(a) Input a key word of 6 digits or less in hexadecimal.

(b) When a key word is not set, a blank is displayed.

(c) To cancel a key word, press the [CLEAR] → [GO] keys.

(9) If writing to ACPU has not been executed and the mode is transferred to the OTHERS mode after operating "2: PARAM. SETTING", the parameters in the A8PUE return to the setting before operating "2: PARAM. SETTING".

## 6.9 OTHERS (O) Mode Operations

This mode changes the set values of timers and counters, does PC error checks, PC memory all clears, as well as setting, etc. of the A8PUE.
An item can be selected in the menu format in the OTHERS mode, and the following operation can be executed:

OTHERS mode

- T/C set values change ..... Changing the T/C set values in the program.
- PC check
  - Error step read..... An error description occurred in an ACPU and the error step is checked.
  - Program check..... Presence/absence of the duplex coil/instruction code/END instruction in the program is checked.
- PC system
  - Monitoring
    - Link monitoring ..... The link state of the MELSECNET(II)/B is checked.
    - Buffer memory batch monitoring ..... The buffer memory contents of a special-function module are checked.
    - Clock monitoring..... The ACPU clock is checked.
  - All clear
    - PC memory all clear..... All clear is done for the ACPU memory.
    - Program all clear ..... All sequence programs operated by ACPU are cleared.
    - Device memory all clear..... All device memories except the special D, special M, and R of an ACPU are cleared.
  - Switching
    - PC number setting..... Enables operations of another station's ACPU on MELSECNET(II)/B. (Switching of ACPUs)
    - Main/sub switching..... Switches the sequence program to be operated.
  - Others
    - Remote RUN/STOP ..... The execution status of ACPU is switched forcibly.
    - Machine language reading and writing ..... Machine language reading and writing is executed from/to the ACPU memory.
- PU setting
  - Program mode selection..... Write enabled/disabled when the ACPU is in the RUN state is set.
    Sets whether only the MONITOR and TEST modes are used.
  - Conductivity state display ..... Sets whether or not the conductivity status of each instruction is displayed when the list monitoring function is used.
  - Buzzer setting ..... ON/OFF of the buzzer when an A key is input is set.

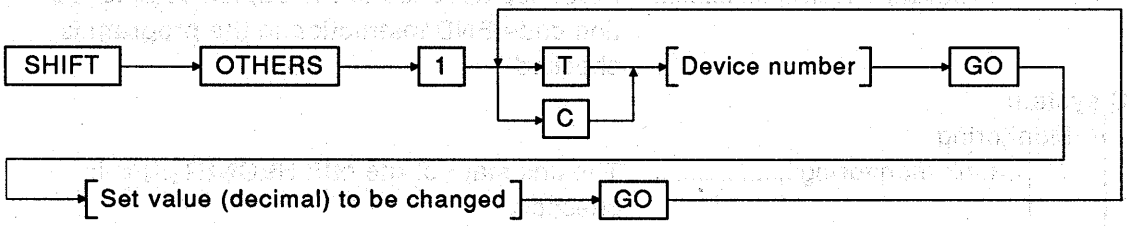
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

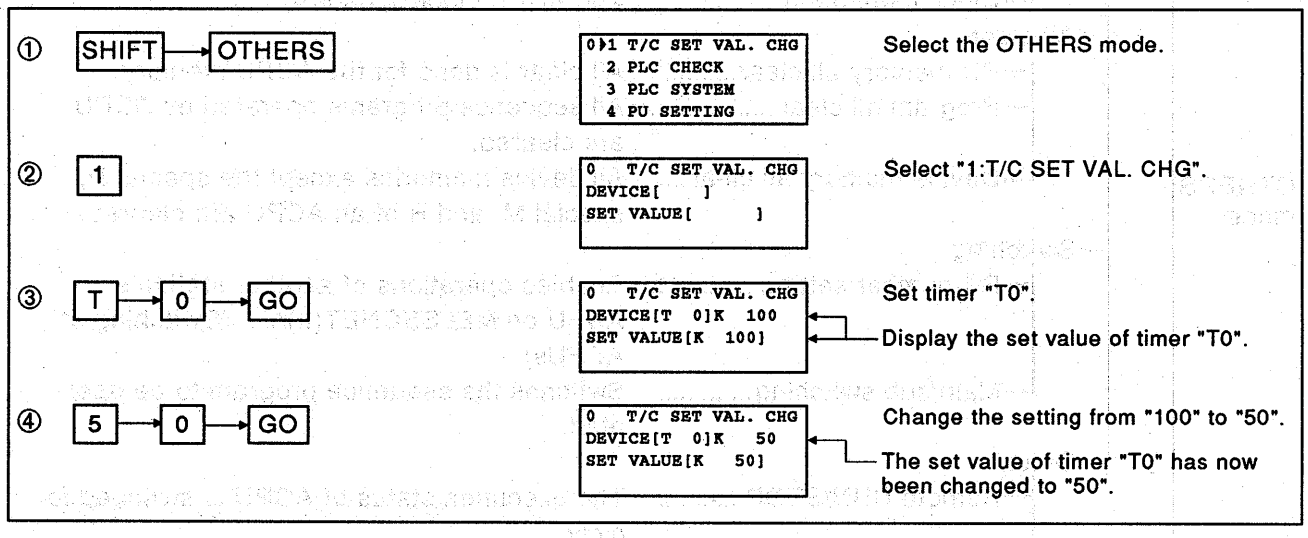
### 6.9.1 Changing the T/C set values when the ACPU is in the RUN state

This operation changes the set value (the constant designation) of T (timer)/C (counter) when the connected ACPU is in the RUN state.
In addition, when the indirectly designated set values of T and C (designated by device D) are changed, changing these set values follows changing the TEST mode current value (see Sections 6.7.2 and 6.7.4).

**[Basic operation]**




**[Sample operation]**



**[Explanation]**

(1) Always change the set value of a constant in decimal.
It can be changed from a constant to indirect designation, and it cannot be changed from indirect designation to a constant.

(2) Pressing the [END] or [CLEAR] → [CLEAR] keys returns to the OTHERS mode menu.

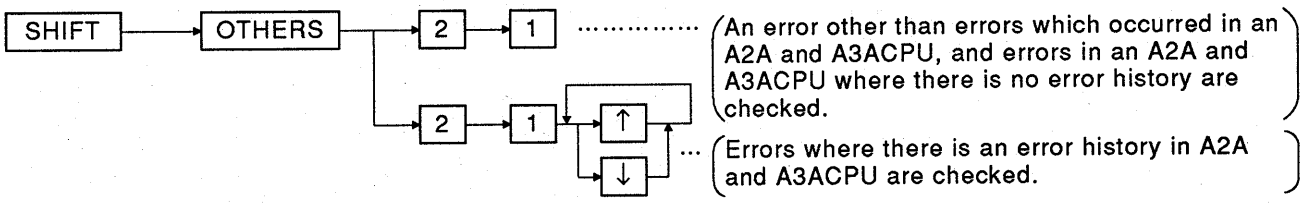
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.9.2 Checking an error step/error code When an error occurs (Error check)

This operation checks the error step number/error code of the error that occurs in an ACPU.

**[Basic operation]**


SHIFT
OTHERS
2
1
2
1
An error other than errors which occurred in an A2A and A3ACPU, and errors in an A2A and A3ACPU where there is no error history are checked.
Errors where there is an error history in A2A and A3ACPU are checked.

**[Sample operation]**

(1) In the case of ACPUs other than an A2A or A3ACPU

① SHIFT → OTHERS

```
0▶1 T/C SET VAL. CHG
  2 PLC CHECK
  3 PLC SYSTEM
  4 PU SETTING
```

Select the OTHERS mode.

② 2

```
0▶1 READ ERR. STEP
  2 PROGRAM CHECK
```

Select "2:PLC CHECK".

③ 1

```
0 ERR. STEP =  25
  SP.UNIT DOWN
  ERR. CODE =  12
```

- Select "READ ERR. STEP".
- The error description is displayed.

[When there is not an error]

```
0 ERR. STEP =   0
  ERROR NOT FOUND
  ERR. CODE =   0
```

(2) When there are no error histories in an A2A or A3ACPU

| | | | |
|---|---|---|---|
| ① | SHIFT → OTHERS | 0▶1 T/C SET VAL. CHG<br>2 PLC CHECK<br>3 PLC SYSTEM<br>4 PU SETTING | Select the OTHERS mode. |
| ② | 2 | 0▶1 READ ERR. STEP<br>2 PROGRAM CHECK | Select "2:PLC CHECK". |
| ③ | 1 | 0 ERR. STEP = 25<br>SP.UNIT DOWN<br>ERR. CODE = 41<br>ERR. INFO = 411 | • Select "1:READ ERR. STEP".<br>• The error description is displayed. |

(3) When there are error histories in an A2A or A3ACPU

| | | | |
|---|---|---|---|
| ① | SHIFT → OTHERS | 0▶1 T/C SET VAL. CHG<br>2 PLC CHECK<br>3 PLC SYSTEM<br>4 PU SETTING | Select the OTHERS mode. |
| ② | 2 | 0▶1 READ ERR. STEP<br>2 PROGRAM CHECK | Select "2:PLC CHECK". |
| ③ | 1 | 0 ERR. STEP = 25<br>SP.UNIT DOWN<br>ERR. CODE = 41<br>↓ ERR. INFO = 411 | Select "1:READ ERR.STEP".<br>The newest error description is displayed. |
| ④ | ↓ | 0 ERR. STEP = 25<br>1SP.UNIT DOWN<br>ERR. CODE = 41 411<br>↓ 92/01/10 10:57:30 | The error description of error history number 1 is displayed. |
| ⑤ | ↓ | 0 ERR. STEP = 0<br>2AC DOWN<br>ERR. CODE = 9 0<br>↓ 92/01/20 11:55:10 | The error description of error history number 2 is displayed. |

**[Explanation]**

(1) Section 7.2 gives details about error indications.

(2) If an A2A or A3ACPU has an error history, errors that occurred are displayed in order from the latest error by pressing the [ ↓ ] key.

(3) The ACPU Programming Manual (Common Instructions) and the User's Manual of the connected CPU give details about error codes and error messages.

(4) Pressing the [END] or [CLEAR] key returns to the "2:PLC CHECK" menu.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

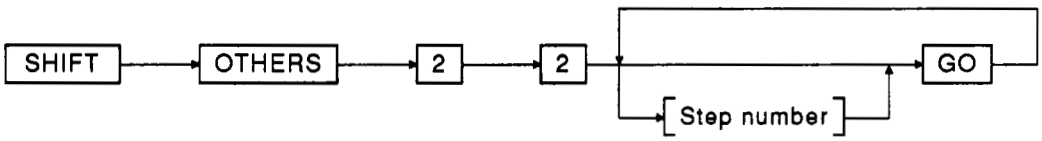
| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.9.3 Checking a program

This operation executes the following checks of sequence programs written to an ACPU.

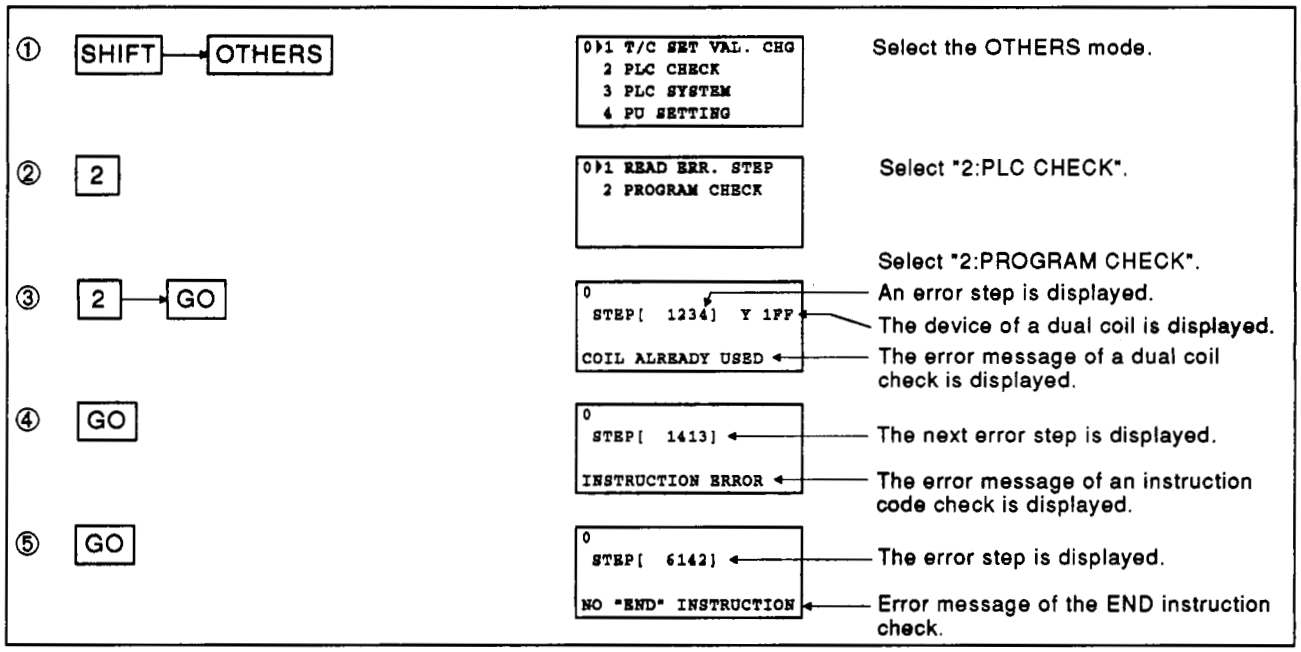
1) Dual coil check ..... This checks whether the device (Y, M, L, B, F, T, C) used in a sequence instruction (OUT, SET, SFT, PLS, MC, PLF) is a dual coil.
2) Instruction code check ..... This checks whether or not there is an error in the instruction codes (memory contents) of a sequence program. (The arrangement of instructions is not checked.)
3) END instruction check ..... This checks whether or not there is an END instruction in the sequence program.

**[Basic operation]**




**[Sample operation]**

If there are a dual coil, an instruction code error, or an END instruction missing error



**[Explanation]**

(1) The A8PUE starts a program check and checks "dual coil", "instruction code error", and "END instruction missing" of all instructions in the range from a designated step to the END instruction of a sequence program written to an ACPU. (Therefore, checking will take some time.)

(2) When a program check is completed normally, the step number of the END instruction and "END" are displayed.

(3) When an error or fault is detected by a program check, the step number where the error or fault occurred and the message that corresponds to the fault contents are displayed, and the program check is interrupted.

1) Press the [GO] key to execute continuous program checking. A program check is started from the step next to a stopped step.

2) When modifying error or fault contents detected by program check, press the [READ] key, or [SHIFT] → [WRITE] keys when an error or fault is detected (when a program check is interrupted).
The program of the step number where an error or fault occurred is displayed by pressing the [READ] key, or [SHIFT] → [WRITE] keys.

Therefore, correction can be done easily. When an error or fault occurs, modify the sequence program as follows:

Designated dual coil ..... Modify the program if necessary.
Instruction code error ..... Overwrite the same instruction. (Rewrite)
END instruction missing ..... Add it.

3) Use the following methods to continue the program check after modifying the program:

Method of continuing a program check from the stopped step
..... Select "2:PROGRAM CHECK" from the OTHERS mode, and press the [GO] key. (The interrupted number of steps is displayed.)
Method of continuing from the designated step
..... Select "2:PROGRAM CHECK" from the OTHERS mode, input the designated step number, and press the [GO] key.

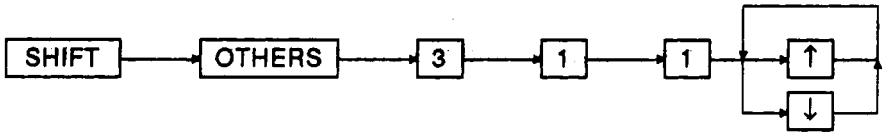
(4) Pressing the [END] or [CLEAR] key returns to the "2: PLC CHECK" menu.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | EPROM | |

### 6.9.4 Monitoring the MELSECNET(II)/B link state (link monitoring)

This operation monitors the MELSECNET(II)/B link state of the self.
THE Reference Manual of data link system gives details about the display contents.

**[Basic operation]**




**[Sample operation]**

Link monitoring is executed when the connected station is a master station (M/m)

| | | | |
|---|---|---|---|
| ① | SHIFT → OTHERS | 0➧1 T/C SET VAL. CHG<br>2 PLC CHECK<br>3 PLC SYSTEM<br>4 PU SETTING | Select the OTHERS mode. |
| ② | 3 | 0➧1 MONITOR<br>2 ALL CLEAR<br>3 PC No. MAIN/SUB<br>4 OTHERS | Select "3:PLC SYSTEM". |
| ③ | 1 | 0➧1 LINK MONITOR<br>2 BUFFER MEM. MON<br>3 CLOCK MONITOR | Select "1:MONITOR". |
| ④ | 1 | 0 SCAN TIME (ms)<br>F LOOP:OK MAX. 20<br>R LOOP:OK MIN. 10<br>PRESENT 20 | Select "1 LINK MONITOR", and the loop state and link scan time are displayed. |
| ⑤ | ↓ | 0[L 1] 10 PARAM.:0<br>CYCLIC COMM.OTHER:0<br>CPU RUN F LOOP:0<br>R LOOP:0 | Pressing the [ ↓ ] key displays the state of machine No. 1.<br>(Example of displaying a local sta tion) |
| ⑥ | ↓ | 0[R 2] 10 PARAM.:0<br>CYCLIC COMM.OTHER:0<br>CPU RUN F LOOP:0<br>R LOOP:0 | Pressing the [ ↓ ] key displays the state of machine No. 2.<br>(Example of displaying a remote I/O station) |

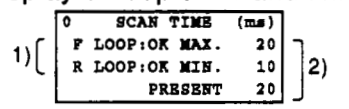
**[Explanation]**

(1) The MELSECNET(II)/B link state can be displayed on only the self. Even if it is set at another station by the PC number setting, the MELSECNET(II)/B link state of the self is displayed.

(2) Link state display contents differ depending on whether the connected station is a master station, local station, or remote I/O station. However, when connecting to a tier-3 master station, it is displayed as a master station. A local station cannot be displayed.

(3) When connecting to a master station the loop state and link scan time as well as the operating state of a slave station can be displayed.

(a) Display of loop state and link scan time

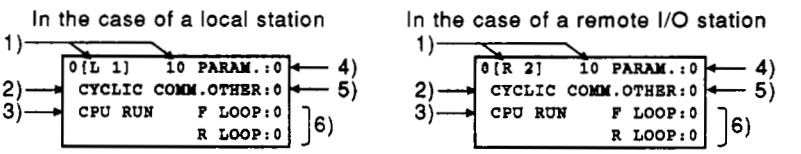



1) Positive loop line (F loop) or reverse loop line (R loop) state of a master station is displayed.
(The contents of the special relay for links M9225 and M9226 are displayed.)

OK: When the line is normal
NG: When the line is abnormal

2) Max., min., and current values of a link scan time are displayed. The contents of special registers for link D9207 to D9209 are displayed.

(b) Operating state of a slave station




1) The station number during monitoring and the number of connected slave stations are displayed.
(The contents of the special registers for link D9243 and D9244 are displayed.)

[Ln] : In the case of a local station
[Rn] : In the case of a remote I/O station
m : Number of connected slave stations

2) The following messages display communications states:
(The contents of the special registers for link D9224 to D9231 are displayed.)

CYCLIC COMM : When normal communications has been executed
DISCONNECTED : When communications has been interrupted or is in a disconnected state
PARAM. COMM : When parameters is communicated with a master station

3) The operating state of the ACPU of a monitoring slave station is displayed by the following message:
(The contents of the special registers for link D9212 to D9215 are displayed.)

CPU RUN : When the ACPU is in the RUN state.
CPU STOP : When the ACPU is in the STOP or PAUSE state.

4) When the monitoring slave station is 3-tier master station, it is displayed on a remote I/O station whether an I/O allocation is correct or not.
(The contents of the special registers for link D9220 to D9223 are displayed.)

0 : Normal
E : Error

5) It displays whether the monitoring local station could find that an error occurred in other local stations.
(The contents of the special registers for link D9216 to D9219 are displayed.)

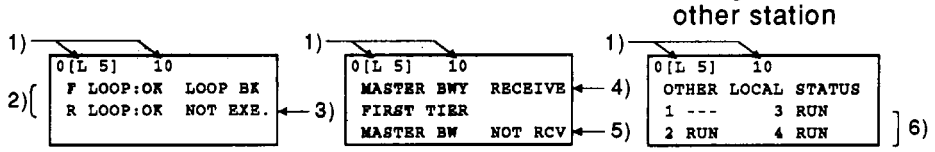
0 : Normal
E : Error
"0" is displayed When the monitoring station is a remote I/O station.

6) The state of a positive loop line (F loop) or the reverse loop line (R loop) of a monitoring slave station is displayed.
(The contents of the special registers for link D9232 to D9239 are displayed.)

0: Normal
E: Error

(4) When it is connected to a local station, the loop state, BWY receive state, and the operating state of other slave stations can be displayed.

(a) Loop state display (b) BWY receive status display (c) Display of the operating state of another station




1) The station number of the self and the number of connected slave stations are displayed.
(The contents of the special registers for link D9243 and D9244 are displayed.)

[Ln] : In the case of a local station
[Rn] : In the case of a remote I/O station
m : Number of connected slave stations

2) The state of the positive loop line (F loop) or the reverse loop line (R loop) of the self is displayed.
(The contents of the special relay for link M9241 and M9242 are displayed.)

OK: When the line is normal
NG: When the line is abnormal

3) It displays whether or not loopback is executed in the self.
(The contents of the special relay for link M9243 are displayed.)

EXECUTED : When executing a loopback in the self
NOT EXE. : When not executing a loopback in the self

4) It displays whether or not data of link relay (B), link register (W), or link output (Y) could be received from the master station.
(The contents of the special relay for link D9246 are displayed.)

RECEIVED : When receiving B, W, or Y of a master station by using cyclic communications
NOT RCV : When the self cannot receive B, W, and Y of a master station because of disconnection.

5) It displays whether or not the link relay (B) and link register (W) can be received from the master station of a higher loop.
(The contents of the special relay for link D9247 are displayed.)

RECEIVED : When receiving B, W, and Y of a master station using cyclic communications
NOT RCV : When B and W of the master station of a tier-2 system cannot be received, or when turning ON M9208

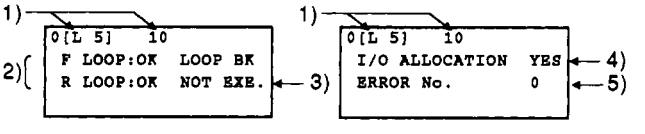
6) The operating state of other local stations is displayed by the following messages:
(The contents of the special registers for link D9248 to D9255 are displayed.)

RUN : When the ACPU is in the RUN state.
STOP : When the ACPU is in the STOP or PAUSE state.
DOWN : When it goes into the disconnected state by the power supply going OFF.

When the monitoring station is a remote I/O station, the display remains "RUN".

(5) When connected to a remote I/O station, the loop state, I/O allocation state, and error codes can be displayed.

(a)Loop state display (b) I/O allocation state and an error code displays



1) The station number of the self and the number of connected slave stations are displayed.
(The contents of the special registers for link D9243 and D9244 are displayed.)

[Ln] : In the case of a local station
[Rn] : In the case of a remote I/O station
m : Number of connected slave stations

2) The state of a positive loop line (F loop) or the reverse loop (R loop) of the self is displayed.
(The contents of the special relay for link M9241 and M9242 are displayed.)

OK: When the line is normal
NG: When the line abnormal

3) Displays whether or not a loopback is executed in the self.
(The contents of the special relay for link M9243 are displayed.)

EXECUTED : When executing a loopback in the self
NOT EXE. : When not executing a loopback in the self

4) Displays whether or not I/O allocation is being executed in a master station.

YES : When I/O allocation is executed in a master station
NO : When I/O allocation is not being executed in a master station

5) The error code that is occurring in the self is displayed.

(6) When a connected station is not performing data link, either of the following message is displayed according to the state:

| | |
|---|---|
| WAITING PARAM. | : When waiting for the receive of parameter information from the master station |
| DISCONNECTED | : When the self goes into the disconnected state and communications is interrupted |
| OFF-LINE | : When the mode of the self is set to off-line, self-loopback test, or station-to-station test |
| LOOP TEST | : When the mode of the self is set to the positive loop test or the reverse loop test |

(7) Pressing the [END] or [CLEAR] keys returns to the "1:MONITOR" menu.

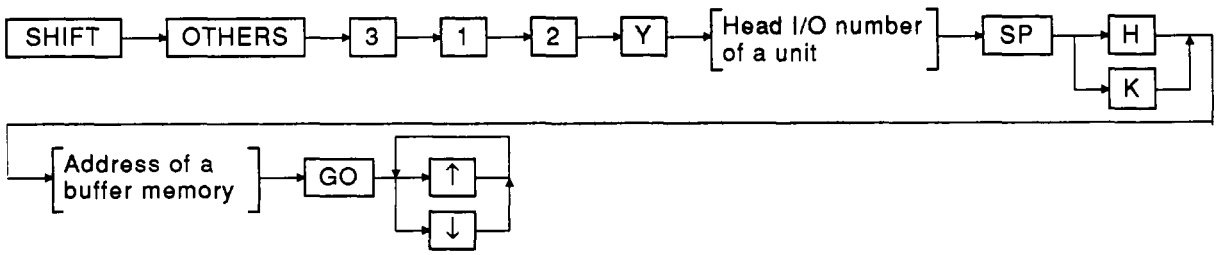
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.9.5 Monitoring the buffer memory of a special-function module (Buffer memory batch monitoring)

This operation monitors the buffer memory contents of a special-function module.
The Manual of the monitoring special-function module gives details about display contents.

**[Basic operation]**




**[Sample operation]**

When the digital output value (buffer memory address: 10) and the I/O numbers X/Y0B0 to 0CF installed in an A68AD are monitored

| | | | |
|---|---|---|---|
| ① | SHIFT → OTHERS | 0▶1 T/C SET VAL. CHG<br>2 PLC CHECK<br>3 PLC SYSTEM<br>4 PU SETTING | Select the OTHERS mode. |
| ② | 3 | 0▶1 MONITOR<br>2 ALL CLEAR<br>3 PC No. MAIN/SUB<br>4 OTHERS | Select "3:PLC SYSTEM". |
| ③ | 1 | 0▶1 LINK MONITOR<br>2 BUFFER MEM. MON<br>3 CLOCK MONITOR | Select "1:MONITOR". |
| ④ | 2 | 0 BUFFER MEM. MON.<br>KEY IN BUFF.ADDRESS | Select "2:BUFFER MEM. MON". |
| ⑤ | Y → B → SP → K → 1 → 0 | 0 BUFFER MEM. MON.<br>KEY IN BUFF.ADDRESS<br>YB K10 | Head I/O number X/Y0B0 of a unit Input the first 2 digits of the hexadecimal 3 digits, and designate address 10 of the buffer memory. |
| ⑥ | GO | 0 10 1020<br>11 1018<br>12 0<br>13 1114 | 10 to 13 of the buffer memory is monitored. |
| ⑦ | ↓ | 0 14 1116<br>15 1014<br>16 998<br>17 1120 | The following address is monitored by pressing the [ ↓ ] key: |

**[Explanation]**

(1) The special-function head I/O number and the buffer memory address are designated and is monitored.

(a) Designate the head I/O number in the first 2 digits when the head I/O number of the special-function module is represented in 3 digits.

(b) The head address of a buffer memory can be designated in decimal or hexadecimal.
The buffer memory monitoring address display is decimal if the head address is designated in decimal. If it is designated in hexadecimal, the address is displayed in hexadecimal.

(2) When a special-function module is not installed in the designated head I/O number or when a designated address does not have a buffer memory, an address error is displayed.

(3) The display format of a value can be changed by pressing the [HELP] key. (Hexadecimal/octal/ASCII/decimal)
When changing a display format, do so in accordance with the indicated display contents (see Section 6.6.4).

(4) Pressing the [END] or [CLEAR] keys returns to the "1:MONITOR" menu.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.9.6 Monitoring the clock data of an ACPU (clock monitoring)

This operation monitors clock data (D9025 to D9027) of an ACPU.

**[Basic operation]**

SHIFT → OTHERS → 3 → 1 → 3

**[Operation example]**

When clock data is monitored

| | | | |
|---|---|---|---|
| ① | SHIFT → OTHERS | 0▶1 T/C SET VAL. CHG<br>2 PLC CHECK<br>3 PLC SYSTEM<br>4 PU SETTING | Select the OTHERS mode. |
| ② | 3 | 0▶1 MONITOR<br>2 ALL CLEAR<br>3 PC No. MAIN/SUB<br>4 OTHERS | Select "3:PLC SYSTEM". |
| ③ | 1 | 0▶1 LINK MONITOR<br>2 BUFFER MEM. MON<br>3 CLOCK MONITOR | Select "1:MONITOR". |
| ④ | 3 | 0<br>92/04/17<br>10.35.48 | Select "3:CLOCK MONITOR".<br>Clock data is displayed.<br>Top : Year, month, and day are displayed.<br>Bottom: Hour, min, and sec are displayed. |

**[Explanation]**

(1) The clock data of an ACPU is monitored by using the information in D9025 to D9027 of the connected station.

(2) When monitoring is started and completed, the A8PUE turns M9028 ON and OFF automatically.

(3) Do not perform this operation for an ACPU which doesn't have a clock function.
If clock monitoring is executed for an ACPU without a clock function, an operating error will not occur. However, meaningless data (value of D9025 to D9027) is displayed.

(4) Pressing the [END] or [CLEAR] keys returns to the "1:MONITOR" menu.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | EPROM | |

### 6.9.7 All clearing the memory contents of an ACPU (PC memory all clear)

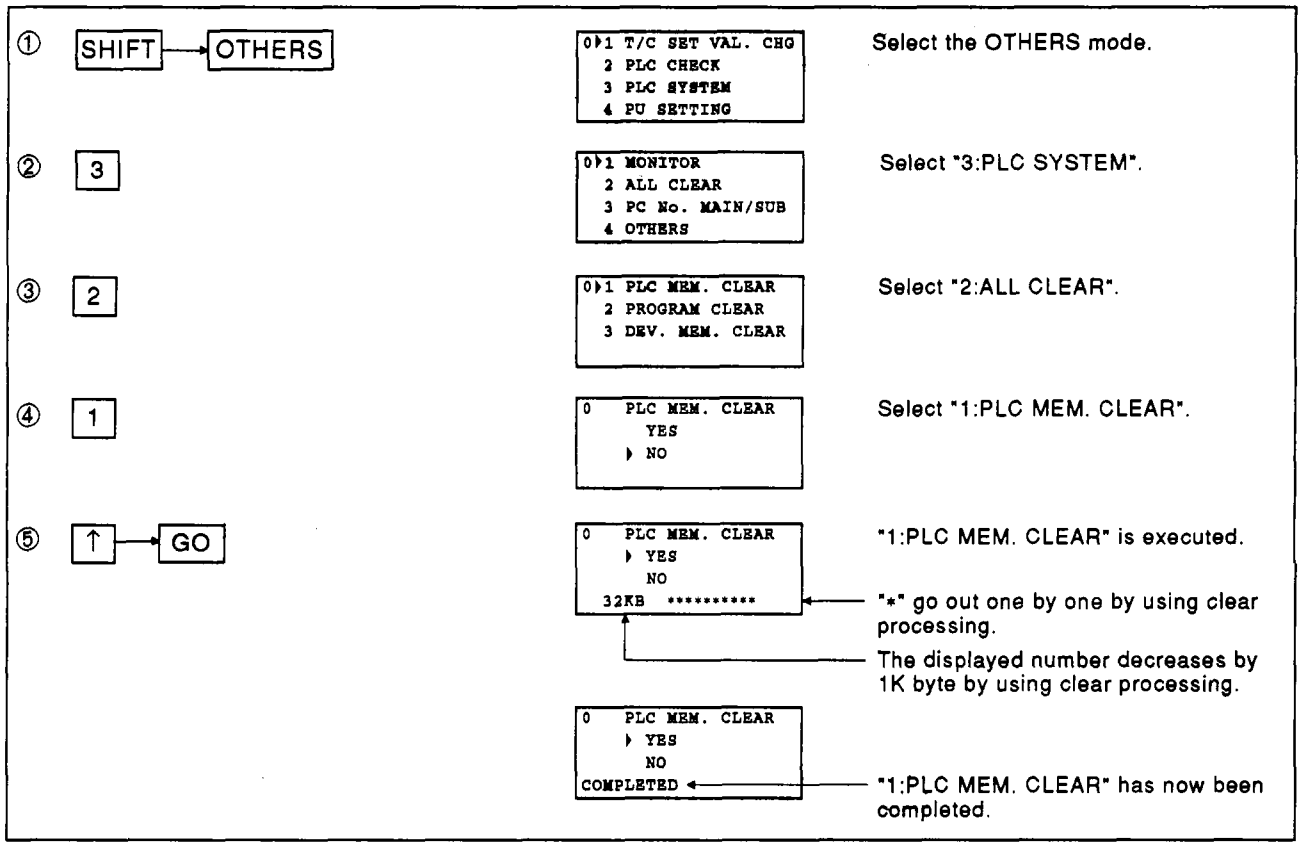
This operation clears all the memory contents (memory cassette) of an ACPU.

**[Basic operation]**

SHIFT → OTHERS → 3 → 2 → 1 → ↑ → GO

**[Sample operation]**

When PC memory all clear is executed




**[Explanation]**

(1) Keep the ACPU in the STOP state.
If it is not in the STOP state, put the ACPU in the STOP state before starting the operation.

(2) When this operation is completed, all contents (memory cassette) of a memory on the ACPU side are cleared, and the parameters are set to default values.
It is necessary to write parameters and sequence programs to the ACPU.

(3) Pressing the [END] or [CLEAR] keys returns to the "2:ALL CLEAR" menu.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.9.8 Clearing sequence programs, microcomputer programs, and T/C set value areas (Program all clear)

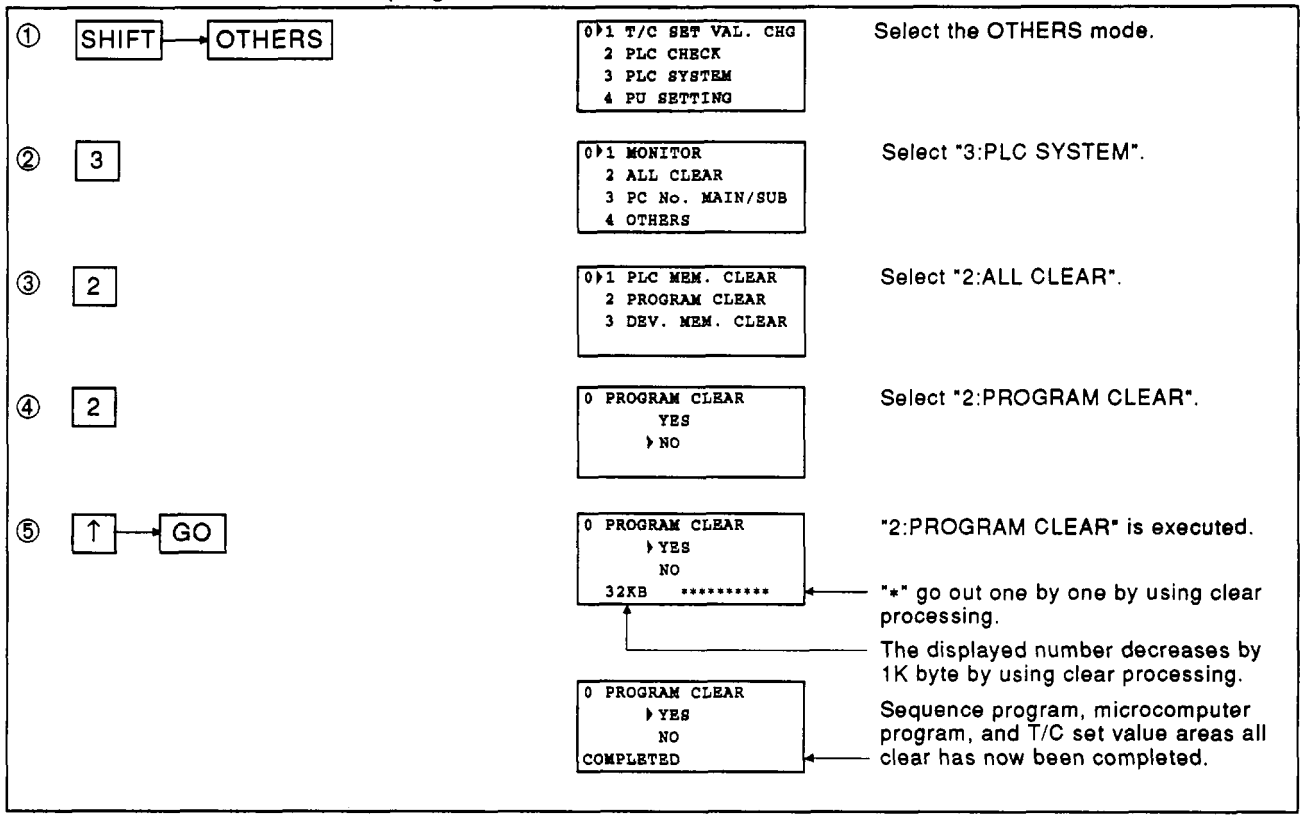
This operation clears the contents of main or sub-sequence programs, microcomputer programs, and T/C set value areas currently being read to the A8PUE.

**[Basic operation]**

SHIFT → OTHERS → 3 → 2 → 2 → ↑ → GO

**[Sample operation]**

When program all clear is executed




**[Explanation]**

(1) Keep the ACPU in the STOP state.
If it is not in the STOP state, put the ACPU in the STOP state before starting the operation.

(2) When this operation is completed, all ACPU sequence program and microcomputer program and T/C set value area contents are cleared.
It is necessary to write parameters and sequence programs to the ACPU.

(3) Pressing the [END] or [CLEAR] keys returns to the "2:ALL CLEAR" menu.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.9.9 Clearing the device memory of an ACPU (Device memory all clear)

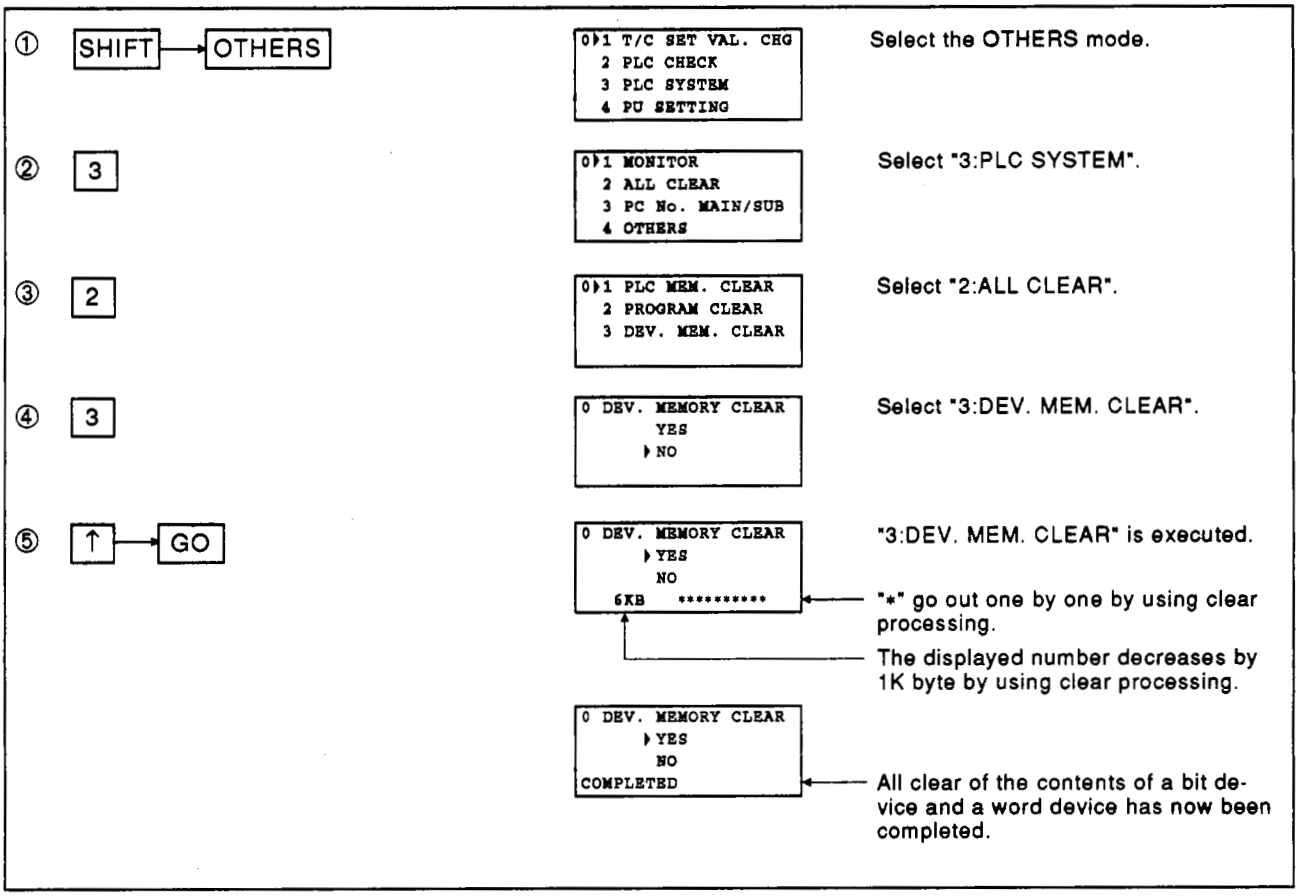
This operation clears the bit devices in the ACPU, and the word device contents (special relay (M), special register (D), and file register (F)).

**[Basic operation]**

SHIFT → OTHERS → 3 → 2 → 3 → ↑ → GO

**[Sample operation]**

When device memory all clear is executed




**[Explanation]**

(1) Keep the ACPU in the STOP state.
If it is not in the STOP state, put the ACPU in the STOP state before starting the operation.

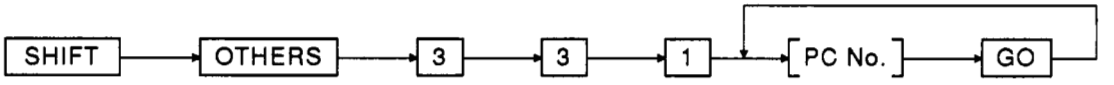
(2) Pressing the [END] or [CLEAR] keys returns to the "2:ALL CLEAR" menu.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | EPROM | |

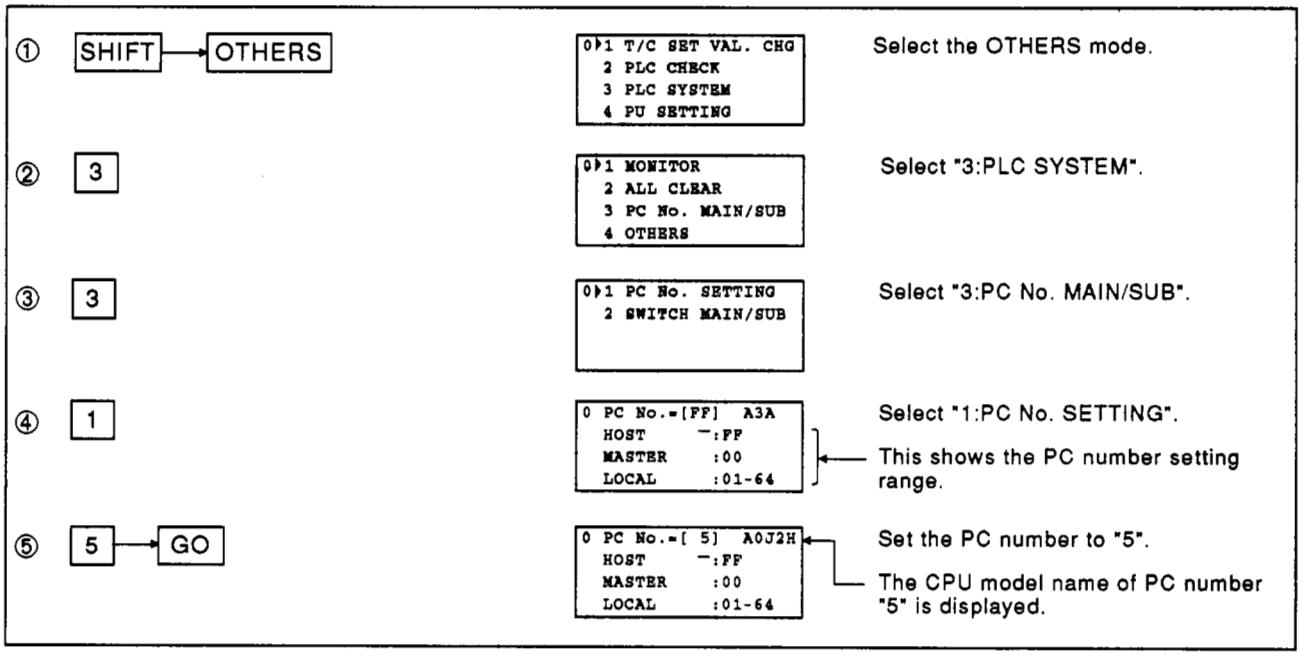
### 6.9.10 Setting the PC number

This operation sets the PC number of the ACPU of the other station that is accessed on MELSECNET(II)/B.
(Default is "FF".)

**[Basic operation]**




**[Sample operation]**




**[Explanation]**

(1) When an entry code is registered in an ACPU, see Section 5.1.3, and input the registered entry code.

(2) Section 5.5 gives details about how to set the PC number.

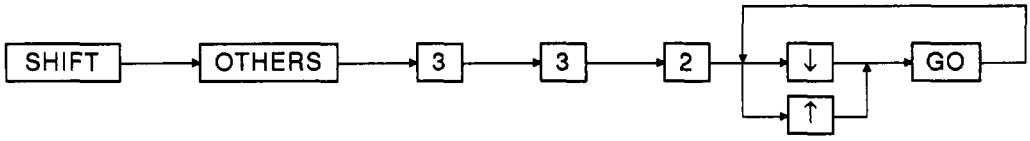
(3) Pressing the [END] or [CLEAR] keys returns to the "3:PC No. SETTING" menu.

| ACPU states | RUN | PAUSE | REMOTE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | UVEPROM | |

### 6.9.11 Switching main/sub-programs

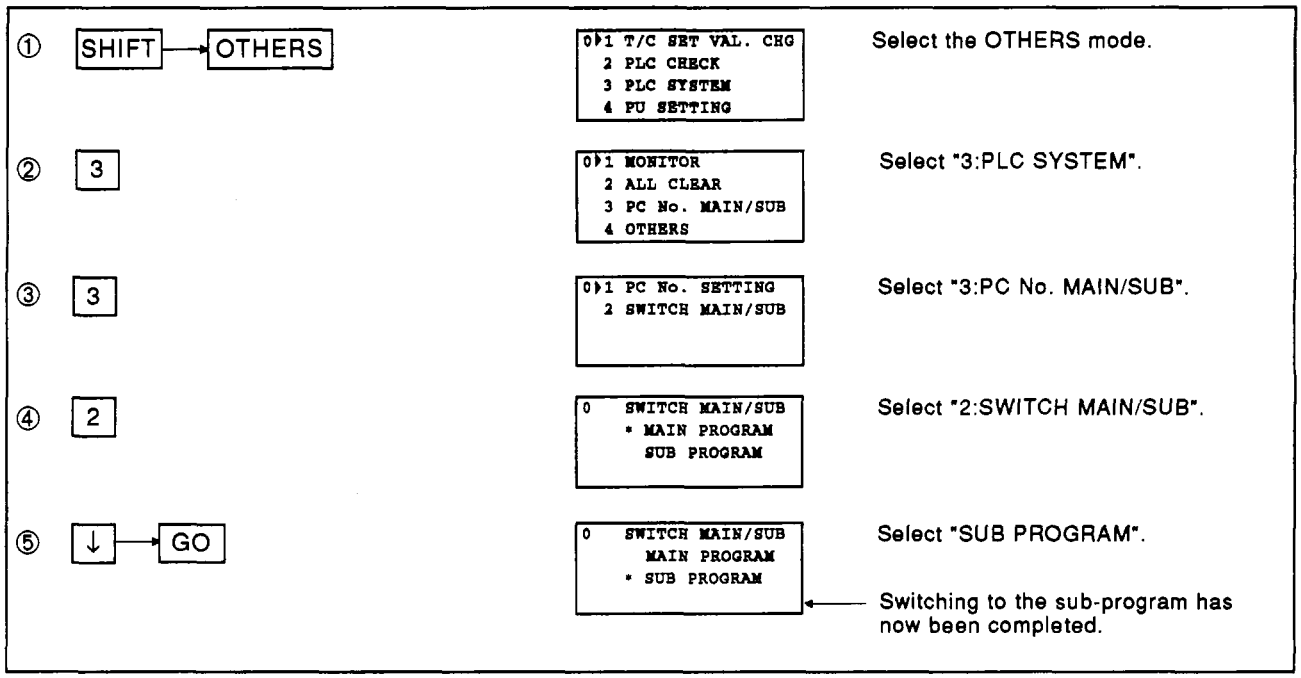
This operation selects the main program/sub-program to be displayed on the A8PUE.
(This operation is available for A3, A3N, A3A, A3H, A3M, A3V, A73 or A373 CPUs.)

**[Basic operation]**




**[Sample operation]**

When switched to a sub-program




**[Explanation]**

Pressing the [END] or [CLEAR] keys returns to the "3:PC No.MAIN/SUB" menu.

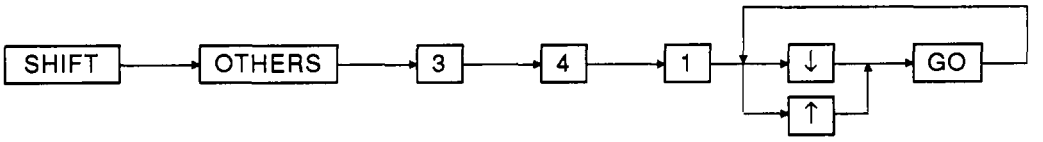
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.9.12 Executing remote RUN/STOP

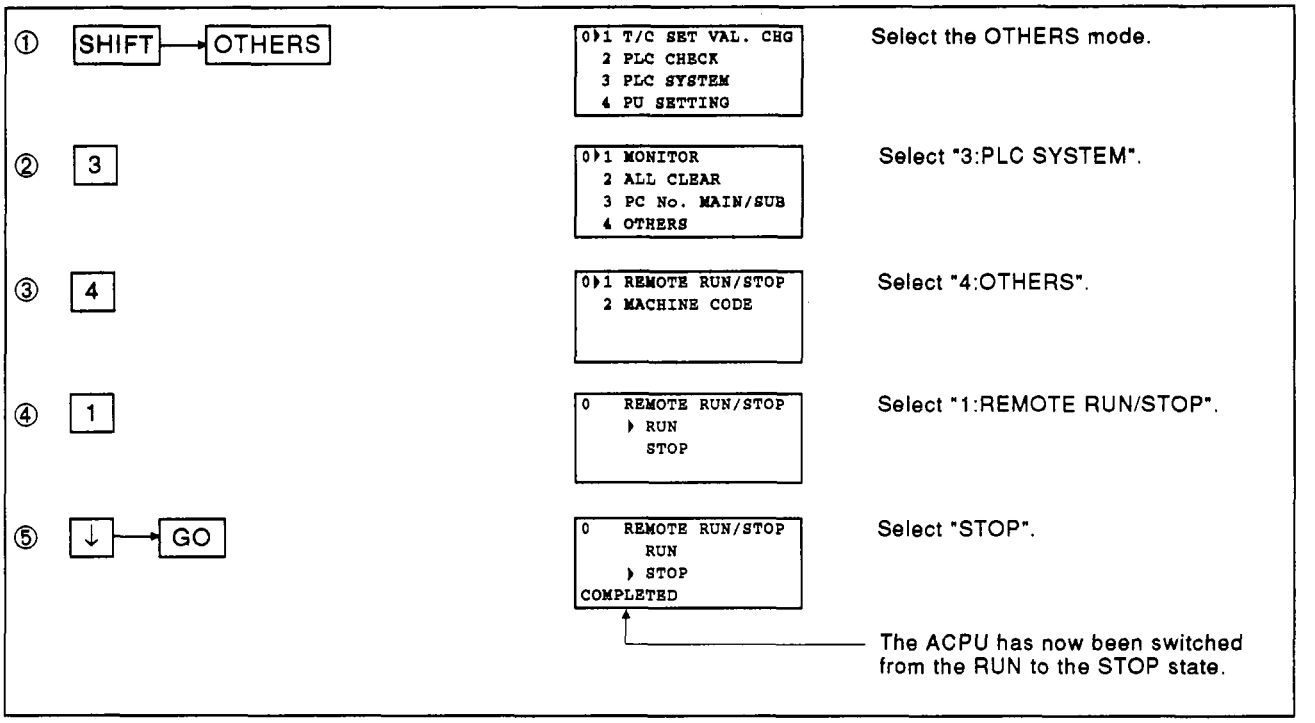
The RUN/STOP state of an ACPU can be operated by the A8PUE.

**[Basic operation]**




**[Sample operation]**

When the ACPU in the RUN state is put in the STOP state from the A8PUE




**[Explanation]**

(1) Begin operations after setting the RUN keyswitch to RUN.

(2) Pressing the [END] or [CLEAR] keys returns to the "4:OTHERS" menu.

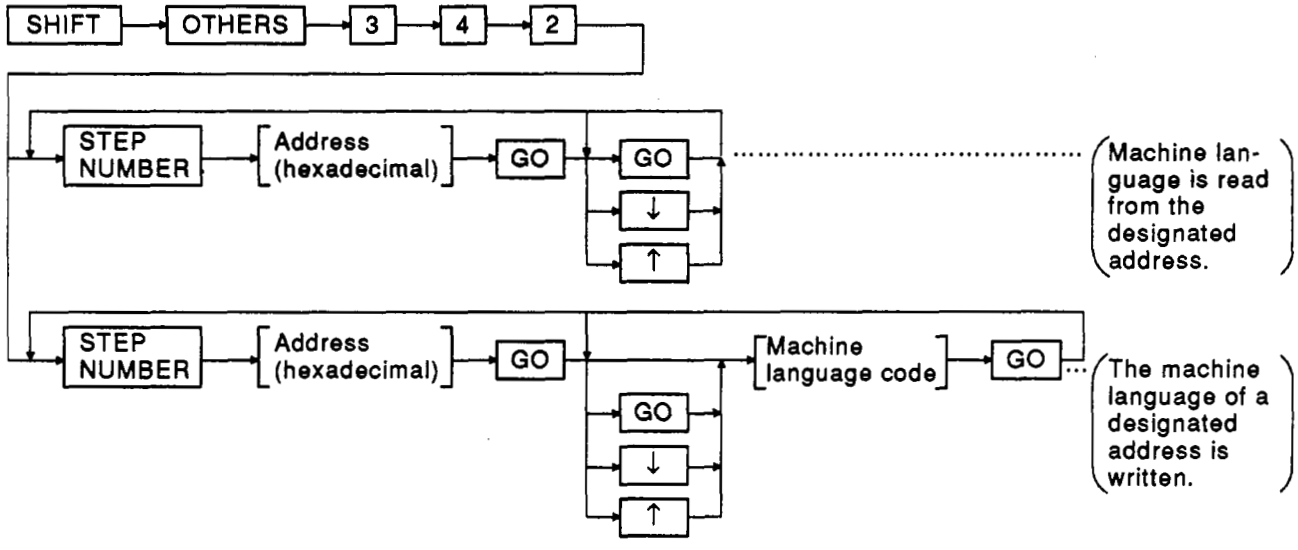
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.9.13 Reading/writing memory contents by using machine language

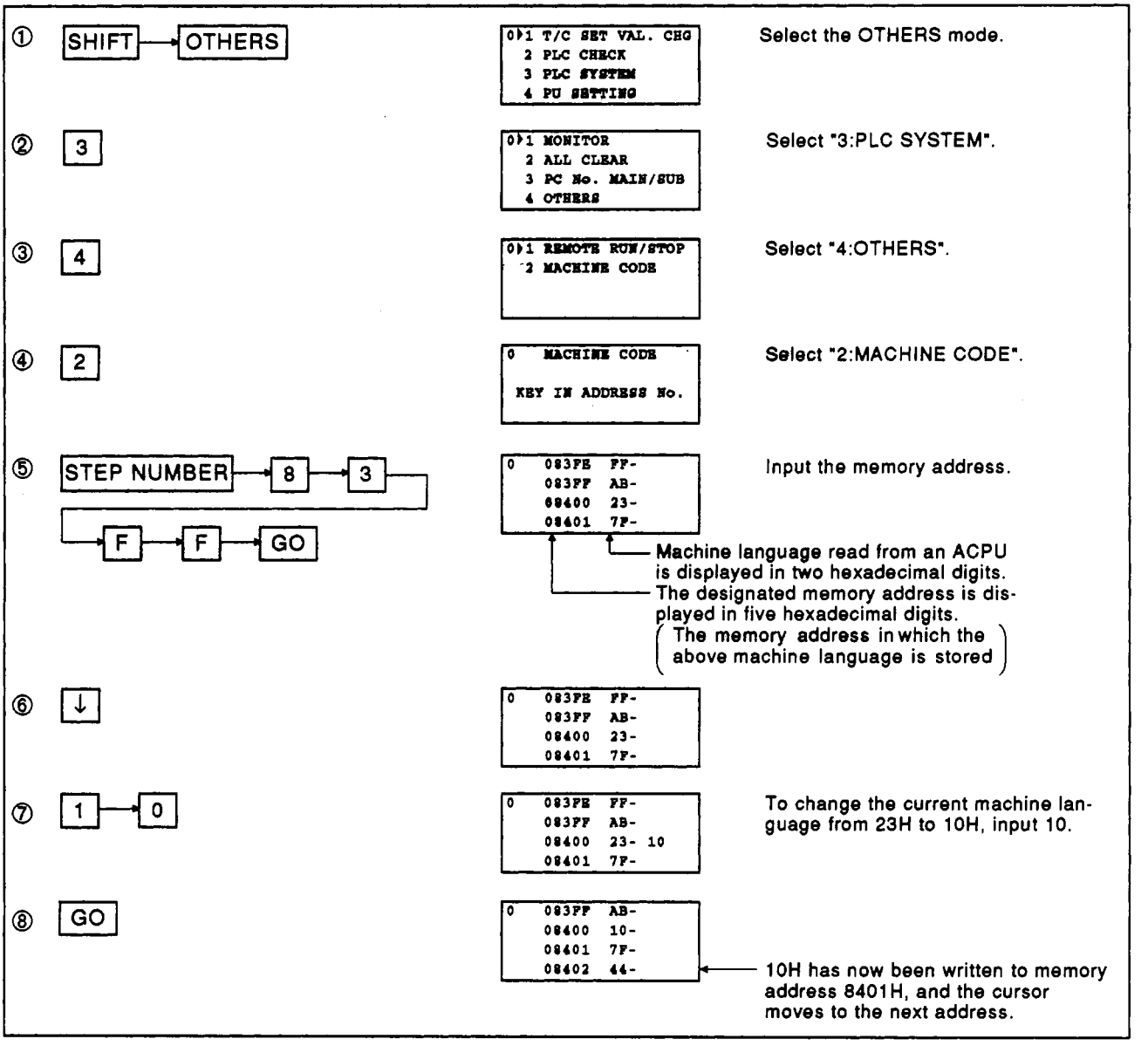
This operation reads memory contents by designating the memory address (absolute address) of an ACPU and does machine language writing to the memory.
Use this function to write/read the user's microcomputer programs to/from an ACPU.

**[Basic operation]**



**[Sample operation]**

When machine language is written to memory address 8400H or after



**[Explanation]**

(1) When writing to the memory is done
Begin operations after putting the ACPU in the STOP state.

(2) When writing to the memory is not done
Operations can be executed whether the ACPU is in the STOP or RUN state.

(3) The address is the memory address (absolute address) that reads the memory contents of an ACPU.
Input five hexadecimal digits max.
(Press the [0] to [9] and [A] to [F] keys on the lower part of the keyboard.)

(4) Input the machine language (two hexadecimal digits max.) to be written to the memory of the currently displayed memory address.
(Use the [0] to [9] and [A] to [F] keys on the lower part of the keyboard.)
When not writing to memory, this operation is unnecessary.
* When the key is pressed, press the correct key continuously.

(5) When writing machine language, execute writing in accordance with the use of the memory to ensure that the ACPU can operate normally.
When unnecessary writing is executed, the ACPU sometimes malfunctions.

(6) Pressing the [END] or [CLEAR] keys returns to the "4:OTHERS" menu.

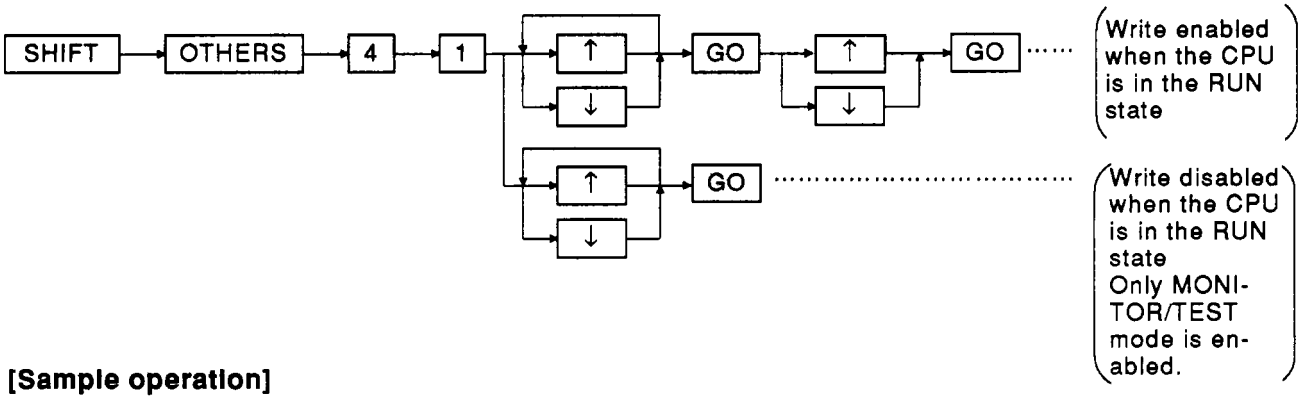
| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|

| Available memory | RAM | EEPROM | EPROM |
|---|---|---|---|

### 6.9.14 Setting write enabled/disabled when the CPU is in the RUN state and setting only MONITOR/TEST mode enabled (Program mode selection)

This operation sets the program write enable/disable when the CPU is in the RUN state from the A8PUE to an ACPU and also sets only MONITOR/TEST mode enabled. (Default is WRITE DISABLED.)

**[Basic operation]**




**[Sample operation]**

When program write is enabled when the CPU is in the RUN state and the CONFIRM NO message is set

| | | | |
|---|---|---|---|
| ① | SHIFT → OTHERS | 0▶1 T/C SET VAL. CHG<br>2 PLC CHECK<br>3 PLC SYSTEM<br>4 PU SETTING | Select the OTHERS mode. |
| ② | 4 | 0▶1 SELECT PRG. MODE<br>2 STATUS DISPLAY<br>3 BEEP SETTING<br>4 EXTENDED MODE | Select "4:PU SETTING". |
| ③ | 1 | 0 SELECT PRG. MODE<br>* WRITE DISABLED<br>WRITE ENABLED<br>MONITOR & TEST | Select "1:SELECT PRG.MODE". |
| ④ | ↓ → GO | 0 SELECT PRG. MODE<br>WRITE ENABLED<br>* CONFIRM YES<br>CONFIRM NO | Select "WRITE ENABLED". |
| ⑤ | ↓ → GO | 0 SELECT PRG. MODE<br>WRITE ENABLED<br>CONFIRM YES<br>* CONFIRM NO | Select "CONFIRM NO". |

**[Explanation]**

(1) When "WRITE DISABLED" is selected
If writing is executed when the CPU is in the RUN state, the "PLC RUN" message is displayed on the fourth line of the screen.

(2) When "WRITE ENABLED: CONFIRM YES" is selected
If writing is executed when the CPU is in the RUN state, the "PLC RUN, PRESS GO KEY TO EXECUTE." message is displayed on the fourth line of the screen.

(3) When "WRITE ENABLED: CONFIRM NO" is selected
Writing when the CPU is in the RUN state can be executed without displaying the message when the CPU is in the RUN state.

(4) When "MONITOR & TEST" is selected
Only the following operations are possible with the monitoring and test function:

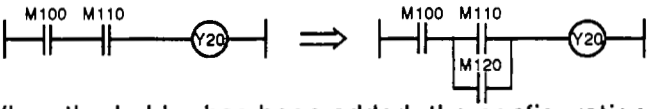
- MONITOR mode.....Section 6.6
- TEST mode.....Section 6.7
- OTHERS mode
  - The T/C set value change.....Section 6.9.1
  - Error step read.....Section 6.9.2
  - Monitoring and switching of the PC system.....Sections 6.9.4, 6.9.5, 6.9.6, 6.9.10, and 6.9.11
  - PU setting.....Sections 6.9.14, 6.9.15, and 6.9.16

**IMPORTANT**

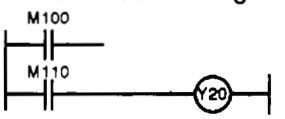
(1) Writing to the EEPROM when the CPU is in the RUN state is disabled.

(2) As for writing when the CPU is in the RUN state, only writing that can be corrected by one instruction can be executed. Be especially careful when the ladder configuration is being changed by writing when the CPU is in the RUN state.

Example)
When a ladder is added by writing when the CPU is in the RUN state




When the ladder has been added, the configuration of a ladder has now been changed as shown below.




(3) When writing when the CPU is in the RUN state is executed, the switching instruction [CHG] for sub-programs and main programs is automatically prohibited. However, an when A3ACPU is used, it is executed.
When writing when the CPU is in the RUN state has been completed, the prohibition of the [CHG] instruction is automatically canceled.

(4) When changing a program by using this function, do not operate the RUN-STOP-PAUSE-STEP RUN switch.
Doing so can destroy the program.

(5) Writing when the CPU is in the RUN state cannot be executed to the PC CPU of another station by setting the PC number on data link system.

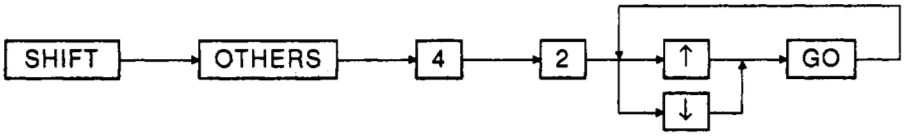
(6) If there is a pulse instruction (PLS,[ ]P) in the program when writing is executed while the CPU is in the RUN state, the CPU executes or does not execute the pulse instruction unconditionally after completing writing when the CPU is in the RUN state. When it is not necessary to execute a pulse instruction, do not change any ladders while the CPU is in the RUN state.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | EPROM | |

### 6.9.15 Setting conductivity display YES/NO

This operation sets the instruction conductivity display YES/NO during list monitoring.
This operation cannot be executed if the connected ACPU is A0J2, A2A, A3H, or A3ACPU.
(Default is the NO conductivity display.)

**[Basic operation]**




**[Sample operation]**

When conductivity is displayed

| | | | |
|---|---|---|---|
| ① | SHIFT → OTHERS | 0▸1 T/C SET VAL. CHG<br>2 PLC CHECK<br>3 PLC SYSTEM<br>4 PU SETTING | Select the OTHERS mode. |
| ② | 4 | 0▸1 SELECT PRG. MODE<br>2 STATUS DISPLAY<br>3 BEEP SETTING<br>4 EXTENDED MODE | Select "4:PU SETTING". |
| ③ | 2 | 0 STATUS DISPLAY<br>YES<br>* NO | Select "2:STATUS DISPLAY". |
| ④ | ↑ → GO | 0 STATUS DISPLAY<br>* YES<br>NO | Select "YES". |

**[Explanation]**

(1) Section 6.6.5 gives details about conductivity display contents.

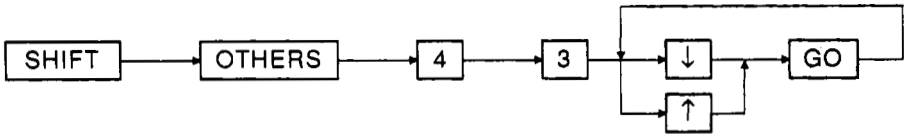
(2) Pressing the [END] or [CLEAR] keys returns to the "4:PU SETTING" menu.

| ACPU states | RUN | STOP | PAUSE | STEP RUN |
|---|---|---|---|---|
| Available memory | RAM | EEPROM | EPROM | |

### 6.9.16 Setting the buzzer ON/OFF when a key is pressed (Buzzer setting)

This operation turns the buzzer ON or OFF when an A8PUE key is pressed. (Default is buzzer ON.)

**[Basic operation]**




**[Sample operation]**

When the buzzer is OFF when an A8PUE key is pressed

| | | | |
|---|---|---|---|
| ① | SHIFT → OTHERS | 0▶1 T/C SET VAL. CHG<br>2 PLC CHECK<br>3 PLC SYSTEM<br>4 PU SETTING | Select the OTHERS mode. |
| ② | 4 | 0▶1 SELECT PRG. MODE<br>2 STATUS DISPLAY<br>3 BEEP SETTING<br>4 EXTENDED MODE | Select "4:PU SETTING". |
| ③ | 3 | 0 BEEP SETTING<br>* ON<br>OFF | Select "3:BEEP SETTING". |
| ④ | ↓ → GO | 0 BEEP SETTING<br>ON<br>* OFF | Select "OFF". |

**[Explanation]**

Pressing the [END] or [CLEAR] keys returns to the "4:PU SETTING" menu.

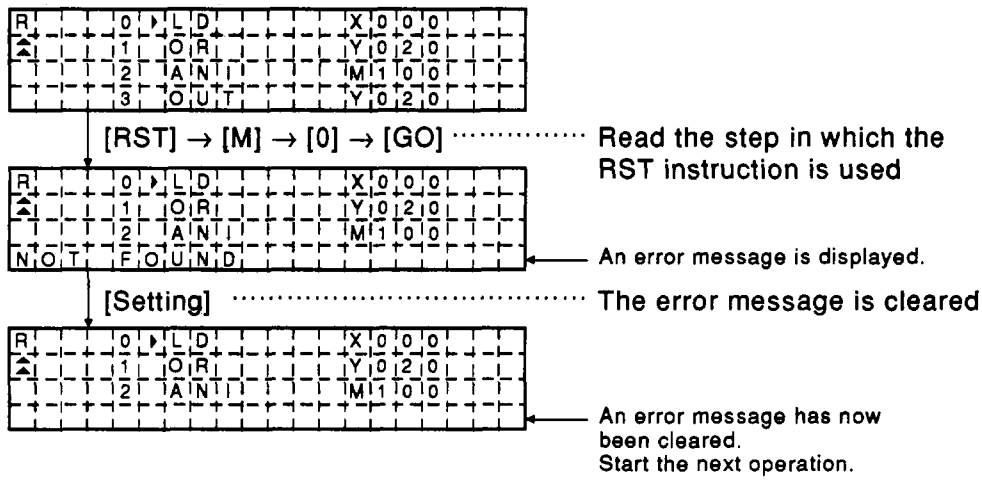
# 7. LISTS OF ERROR MESSAGES

## 7.1 Errors Detected by the A8PUE

Whenever the A8PUE detects an error while operating in any mode, an error message is displayed on the fourth line of the display area.
This section shows the error messages, display states, and corrective actions.
When an error message is displayed, perform the following, and then restart operations.

1) Check the error message.
2) Remove the cause of the error.
3) Press any key
   (doing so clears the error message. Then, the unit returns to the state before the error occurred.)

(Example)


[RST] → [M] → [0] → [GO]
Read the step in which the RST instruction is used
NOT FOUND
An error message is displayed.
[Setting]
The error message is cleared.
An error message has now been cleared.
Start the next operation.

## Table 7.1 List of A8PUE error messages

| Explanation Nos. | Error Message | Display Conditions | Corrective Action(s) |
|---|---|---|---|
| 1 | ADDRESS ERROR | • When doing machine language writing, etc., an address was written to a write-disabled area. | • Set a correct address. |
| 2 | CAN'T BE SELECTED | • During start-up, because of a key word mismatch, an attempt was made to (a) select a mode that cannot be selected, or (b) execute an operation that cannot be done. | • Input the registered key word that resets the A8PUE, and restart operations.<br>• If the registered key word has been forgotten, execute the CPU memory all clear procedure, and restart operations. (See Sections 5.1.3 and 6.9.7.)<br>Remember, that if this procedure is executed, the entire CPU memory will be cleared. |
| 3 | CHECK MEM. CASSETTE | • When clearing a key word or communicating with a CPU during parameter write operations, the memory cassette was either faulty or not installed. | • Replace the faulty memory cassette.<br>• Install a memory cassette correctly. |
| 4 | COIL ALREADY USED | • The same coil was already in the sequence program. | • If it is no problem as far as control is concerned, perform the next operation.<br>• If there is a problem as far as control is concerned, modify the program. |
| 5 | DEVICE ERROR | • The set device symbol was incorrect.<br>• The device number exceeded the range. | • Set a correct device symbol.<br>• Set the number within the CPU device range. |
| 6 | INSTRUCTION ERROR | • Could not be converted into a normal instruction when a program was read. | • When the CPU detects an error, switch the state from RUN to STOP. And then, after resetting the CPU, check the instructions before and after the step where the error occurred, and write a correct instruction. (Section 6.9.2 gives details about checking error steps.) |
| 7 | INSTRUCTION ERROR | • A set instruction was incorrect during READ, WRITE, or INSERT. | • Set a correct instruction. |
| 8 | MEMORY PROTECTED | • When writing was attempted in the WRITE/INSERT/DELETE, etc. modes, the memory protect switch in the memory cassette was ON. | • Set the memory protect switch in the memory cassette to OFF. |
| 9 | MEM. CAP. EXCEEDED | • The memory allocation set in the parameter exceeded the capacity of the memory cassette. | • Reset the parameter within the capacity of the memory cassette. |
| 10 | NO "END" INSTRUCTION | • There was no END instruction. | • Write an END instruction as the last step of the program. |
| 11 | NOT FOUND | • The designation instruction was not found. | • Check the program. |
| 12 | OPERATION ERROR | • An key error was made.<br>• An attempt was made to write data to the ROM or EEPROM of the CPU. | • Operate the key correctly.<br>• Writing to the EEPROM when the CPU is in the STOP state.<br>• Do not attempt to write to the ROM. |
| 13 | PLC COMM. ERR. | • Could not communicate with a CPU normally. | • Retry the operation. If communications still cannot be done, check the following:<br>The connection state of the A8PUE<br>The connection state of the cable<br>The CPU (whether or not an error has occurred) |

| Explanation Nos. | Error Message | Display Conditions | Corrective Action(s) |
|---|---|---|---|
| 14 | PLC RUN | • Write/insert/delete, etc. was attempted when the CPU was in the RUN state. | • Set the CPU in the STOP state.<br>• Reoperate it after selecting WRITE ENABLED by using the program mode selection of the OTHERS mode (see Section 6.9.14). |
| 15 | PLC WRITE ERROR | • Data could not be written to a CPU correctly in the WRITE, INSERT, etc., modes. | • Check the RAM/ROM settings.<br>• Check the installation of the RAM, etc.<br>• Check the setting of the CPU's memory protect switch. |
| 16 | SETTING ERROR | • A set value was not correct. | • Set the value correctly. |
| 17 | STEP NO. EXCEEDED | • A set step number was larger than the maximum step number. | • Set a correct step number. |

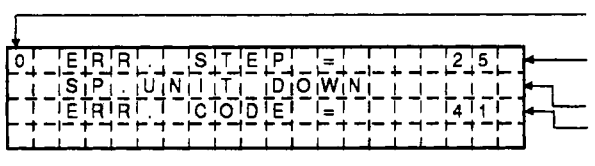
## 7.2 PC CPU Errors

When an error step in the OTHERS mode is read, the error message of the error that is occurring in the ACPU, the error step, etc. are displayed.
This section shows the error messages, error contents, and corrective action(s).
When an error message is displayed, perform the following, and then restart operations:

1) Check the error message.
2) When an error code is not displayed, check the error code of special register D9008 by device monitoring (see Section 6.6.3) in the MONITOR mode.
3) Remove the cause of the error.

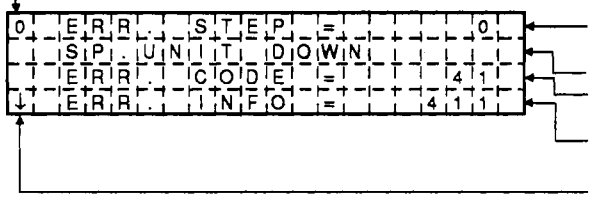
(Display format)

Display example when an error occurs in any CPU but an AnACPU



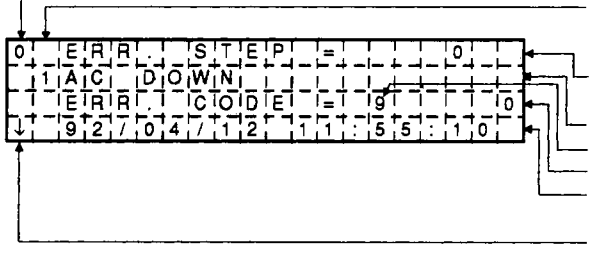

The mode is displayed. (OTHERS mode)
The error occurrence step number is displayed. (D9010 step number)
The error message is displayed.
The error code is displayed. (D9008 error code)

Display example when an error occurs in an AnACPU




The mode is displayed. (OTHERS mode)
The error occurrence step number is displayed (step number D9010).
The error message is displayed.
The error code is displayed. (D9008 error code)
The detailed error code is displayed. (D9091 error code)
Displayed when there is a history of errors.

Example of continuous display when "↓" above is displayed (when there is an error history)




The mode is displayed. (OTHERS mode)
The error history number is displayed.(The ACPU's history management number)
The error occurrence step number is displayed.
The error message is displayed.
The error code is displayed.
The detailed error code is displayed.
The date and time of the error occurrence are displayed.
Displayed when there are other error histories.

**POINT**

When a PC CPU error message is displayed, take corrective action based on the ACPU Programming Manual (Common Instructions) and the User's Manual of each CPU.

## 7.3 Errors When Using the A8PUE in a Link System

The "PLC COMM. ERR." message is sometimes displayed when the A8PUE is used in a link system). When this happens, check the error contents and take corrective action.

| Error Numbers | Error Contents | Corrective Action(s) |
|---|---|---|
| 4 | Processing cancellation : A new processing request was output from the A8PUE though CPU was already processing. | Do key operations from the A8PUE side again. |
| 5 | Sum check error : A link communications sum check error occurred. | This error could be caused by noise influence. Recheck the system. |
| 16 | PC NO. error : The corresponding PC number station does not exist. | Check the PC number setting, and set the correct number. |
| 19 | When the ACPU is reset during monitoring, this error sometimes occurs. | Perform monitoring operations again. |
| 24 | Remote error : Though remote STOP/PAUSE was executed from the computer link unit, etc., remote RUN/STOP was executed from the A8PUE. | Execute remote RUN/STOP/PAUSE from either unit. |
| 32 | Link error : During monitoring of a master station from a slave station, the master station was reset. | Perform monitoring operations again. |
| 34 | EEPROM fault : Writing could not be executed because of a faulty EEPROM. | Replace the EEPROM. |

When error code 25 is displayed, the cause could be one of the following: Check the contents and take corrective action.

(1) When connected to a master station

| Device Numbers | Names | Contents | |
|---|---|---|---|
| M9210 | Link card error (For a master station) | OFF : Normal<br>ON : Abnormal | The link card's H/W is abnormal during control. A link card in the CPU link unit is judged by the CPU. Replace the link unit. |
| M9224 | Link state | OFF : Online<br>ON : Offline, station-to-station test, or self-loopback test | Controlled by the master station state such as offline, station-to-station test, or the self-loopback test mode. Check the mode switch. |
| M9227 | Loop test state | OFF : Not executed<br>ON : A positive loop test or a reverse loop test a being executed. | Controlled by whether a master station is executing a positive loop test or a reverse loop test. |

(2) When connected to a local station

| Device Numbers | Names | Contents | |
|---|---|---|---|
| M9211 | Link card error (For a local station) | OFF : Normal<br>ON : Abnormal | The link card's H/W is abnormal during control. A link card in the CPU link unit is judged by the CPU. Replace the link unit. |
| M9240 | Link state | OFF : Online<br>ON : Offline, station-to-station test, or self-loopback test | Controlled by the local station state such as offline, station-to-station test, or the self-loopback test mode. Check the mode switch. |
| M9257 | Loop test state | OFF : Not executed<br>ON : A positive loop test or a reverse loop test a being executed. | Controlled by whether a local station is executing a positive loop test or a reverse loop test. |

**POINT**

Sometimes an error code that is not stated on the previous page is displayed. When this happens, consult your nearest Mitsubishi representative with details.

# APPENDICES

## APPENDIX 1 COMPARING THE A8PUE WITH THE A7PU/A7PUS

This section shows the main differences between the A8PUE and conventional A7PU/A7PUS programming units.
For more details, see the particular Operating Manuals.

### (1) Comparison of general specifications

| Items \ Models | | A8PUE | A7PU | A7PUS |
|---|---|---|---|---|
| Ambient temperature | When operating | 0 to 40 °C | | |
| | When stored | −20 to 70 °C | −10 to 50 °C | |
| Ambient humidity | When operating | 20 to 85 % RH or less | 10 to 85 % RH or less | |
| | When stored | 10 to 90 % RH or less | | |

### (2) Comparison of performance specifications

| Items \ Models | | A8PUE | A7PU | A7PUS |
|---|---|---|---|---|
| Connection methods | Add-on | A8PUE and A7PU cannot be screwed onto an A1SCPU and A2CCPU. | | A7PUS can only be screwed onto an A1SCPU. |
| | Hand-held | Uses AC30R4-PUS/AC20R4-A8PU cable. | Uses AC30R4/AC300R4 cable. | Uses AC30R4-PUS cable. |
| Display methods | | 20 characters × 4 lines Backlit display | 16 characters × 2 lines Illuminated display | |
| External interfaces | | RS-422I/F × 2, Extension I/F × 1 | RS-422 × 2, Audio cassette I/F × 1 | RS-422I/F × 1 |
| Outside dimensions mm (inch) | Heights | 188 (7.40) | 188 (7.40) | 102 (4.02) |
| | Widths | 95 (3.74) | 79 (3.11) | 109 (4.29) |
| | Depths | 44.5 (1.75) | 44.5 (1.75) | 35.5 (1.40) |

### (3) Comparison of functions

(The operating modes are shown.)

| Items \ Models | A8PUE | A7PU | A7PUS |
|---|---|---|---|
| Applicable CPUs | See Section 3.3.2. | See Section 3.3.2. The device designation for an A2ACPU and an A3ACPU is limited to the same range as an A3HCPU. | |
| PC memory all clear | OTHERS mode/when starting up with a key word | When starting up with a key word | |
| Parameters all clear only | PARAMETER mode | | |
| Programs all clear only | OTHERS mode | WRITE mode (NOP continuous writing) | |

(Continued on the next page.)

(Continued from the previous page.)

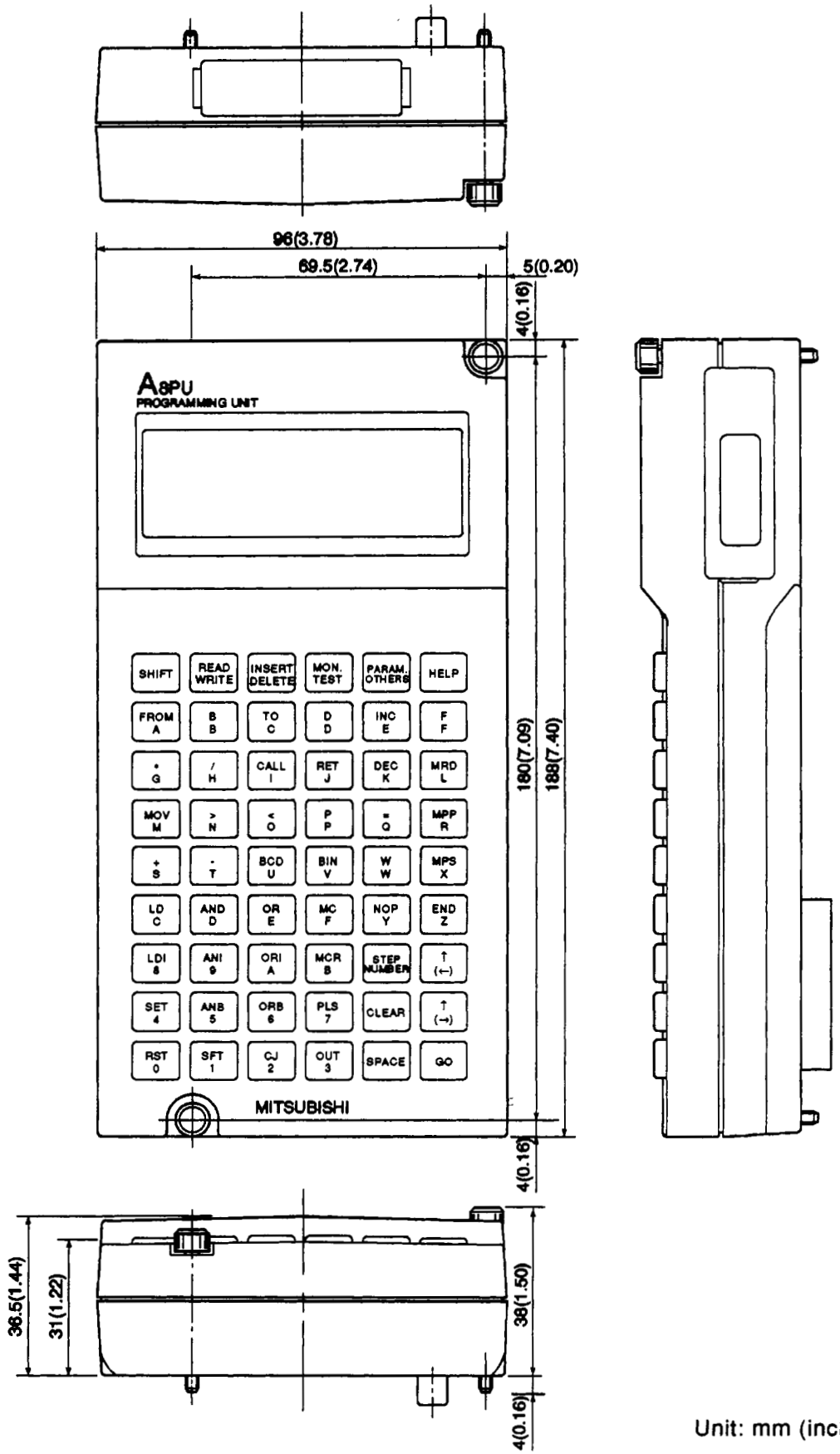
| Items \ Models | A8PUE | A7PU | A7PUS |
|---|---|---|---|
| Devices all clear only. | OTHERS mode | ——— | |
| NOP continuous writing | WRITE mode | | |
| Writing a program | WRITE mode | | |
| T/C set value change: CPU in RUN state | OTHERS mode | WRITE mode | |
| Reading a program | READ mode | | |
| Program scrolling display | READ mode | ——— | |
| Inserting a program | INSERT mode | INSERT/DELETE modes | |
| Moving a program | INSERT mode | ——— | |
| Copying a program | INSERT mode | ——— | |
| Deleting a program | DELETE mode | INSERT/DELETE modes | |
| Deleting a designated range of a program | DELETE mode | ——— | |
| NOP batch delete | DELETE mode | ——— | |
| Monitoring a bit device | MONITOR mode | | |
| Monitoring a word device | MONITOR mode | | |
| Monitoring T/C contact and current position data | MONITOR mode | | |
| Continuity check | MONITOR mode | | |
| Monitoring an offline switch | MONITOR mode | | |
| Searching the output instruction of the designated contact | MONITOR mode | ——— | |
| Switching a numerical value's display format | MONITOR/TEST modes [binary/octal/decimal/hexadecimal/ASCII] | MONITOR mode (decimal/hexadecimal) | |
| Setting/resetting a bit device | TEST mode | | |
| Changing the current value of a word device | TEST mode | | |
| Setting/canceling an offline switch | TEST mode | | |
| Checking a program | OTHERS mode | TEST mode | |
| Error check | OTHERS mode | TEST mode | |
| Reading/writing a machine language | OTHERS mode | TEST mode | |
| Setting parameters | PARAMETER mode | | |
| Setting/changing a key word | PARAMETER mode | | |
| Recording/replaying/verifying for an audio cassette | ——— | Executed by the audio cassette function. | ——— |
| Link monitoring of MELSECNET(II)/B | OTHERS mode | ——— | |
| Monitoring the buffer memory of a special-function module | OTHERS mode | ——— | |
| ACPU clock monitoring | OTHERS mode | ——— | |

(Continued on the next page.)

(Continued from the previous page.)

| Items \ Models | A8PUE | A7PU | A7PUS |
|---|---|---|---|
| Accesing another station | Switching in the OTHERS mode | ——— | |
| Remote RUN/STOP | OTHERS mode | ——— | |
| Setting write enabled/disabled in the RUN state | OTHERS mode | When starting up | |
| Main/sub switching | OTHERS mode | When starting up | |
| Buzzer ON/OFF during key operation | OTHERS mode | ——— | |

## APPENDIX 2 A8PUE OUTSIDE DIMENSIONS




Unit: mm (inch)

## APPENDIX 3 ACPU PARAMETER SETTING SHEET

This section shows the parameters that can be set in different types of ACPUs by the A8PUE.

(1) A0J2CPU parameters that can be set by A8PUE

| | | |
|---|---|---|
| **Latch range setting** | **No latch** | ——— |
| | **1/2 latch** | Latch ranges<br>L1024 to 2047 (L1024 to 1535, S1536 to 2047)<br>T40 to 79/T100 to 119/T124 to 127<br>C64 to 127<br>D256 to 511<br>B200 to 3FF (W200 to 3FF) |
| | **All latch** | Latch ranges<br>L0 to 2047 (L0 to 1535, S1536 to 2047)<br>T0 to 127<br>C0 to 127<br>D0 to 511<br>B000 to 3FF (W000 to 3FF) |
| **Step relay setting** | | Without/with (S1536 to 2047) |

(2) A0J2H, A2C, A1(N), A2(N), and A3(N) CPU parameters that can be set by the A8PUE

| | | | | |
|---|---|---|---|---|
| **Memory capacity** | **Program capacity** | Main sequence: Ksteps | | |
| | | Sub-sequence: Ksteps | | |
| | **File register capacity** | Point ( Kbytes) | | |
| **M, L, and S setting** | | M - | L - | S - |
| **Timer setting** | | Low speed: T - | Medium speed: T - | High speed: T - |
| **Counter setting** | | Counter: C - | Interruption counter: C | |
| **Latch range setting** | | B - | | |
| | | T (low speed) - | T (medium speed) - | T (high speed) - |
| | | C - | D - | W - |
| **WDT setting** | | msec | | |
| **I/O control method** | | Unavailable | | |
| **Key word** | | | | |

(3) A3H and A3MCPU parameters that can be set by the A8PUE

| | | | | |
|---|---|---|---|---|
| **Memory capacity** | **Program capacity** | Main sequence: Ksteps | | |
| | | Sub-sequence: Ksteps | | |
| | **File register capacity** | Point ( Kbytes) | | |
| **M, L, and S setting** | | M - | L - | S - |
| **Timer setting** | | Low speed: T - | Medium speed: T - | High speed: T - |
| **Counter setting** | | Cannot be set | | |
| **Latch range setting** | | B - | | |
| | | T (low speed) - | T (medium speed) - | T (high speed) - |
| | | C - | D - | W - |
| **WDT setting** | | Unavailable (fixed 200 msec) | | |
| **I/O control method** | | Input: | Output: | |
| **Key word** | | | | |

(4) A2A and A3ACPU parameters that can be set by the A8PUE

| | | | | |
|---|---|---|---|---|
| **Memory capacity** | **Program capacity** | Main sequence: Ksteps | | |
| | | Sub-sequence: Ksteps | | |
| | **File register capacity** | Point ( Kbytes) | | |
| **M, L, and S setting** | | M - | L - | S - |
| | | M - | | |
| **Timer setting** | **The number of device points** | | | |
| | **Range of T0 to 255** | Low speed: T | Medium speed: T - | High speed: T - |
| | **Range of T256 to 2047** | Low speed: T | Medium speed: T - | High speed: T - |
| | **Set value storage device** | | | |
| **Counter setting** | **The number of device points** | | | |
| | **Set value storage device** | | | |
| **Latch range setting** | | B - | D - | W - |
| | **Range of T0 to 255** | T (low speed) - | T (medium speed) - | T (high speed) - |
| | **Range of C0 to 255** | C - | | |
| | **Range of T256 to 2027** | T (low speed) - | T (medium speed) - | T (high speed) - |
| | **Range of C256 to 1023** | C - | | |
| **Key word** | | | | |

## APPENDIX 4 ASCII DISPLAY CHARACTER CODE

Higher 4 bits →

| ↓ Lower 4 bits | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | 0 | @ | P | ‘ | p | | | | | | | | |
| 1 | | | ! | 1 | A | Q | a | q | | | | | | | | |
| 2 | | | " | 2 | B | R | b | r | | | | | | | | |
| 3 | | | # | 3 | C | S | c | s | | | | | | | | |
| 4 | | | $ | 4 | D | T | d | t | | | | | | | | |
| 5 | | | % | 5 | E | U | e | u | | | | | | | | |
| 6 | | | & | 6 | F | V | f | v | | | | | | | | |
| 7 | | | ' | 7 | G | W | g | w | | | | | | | | |
| 8 | | | ( | 8 | H | X | h | x | | | | | | | | |
| 9 | | | ) | 9 | I | Y | i | y | | | | | | | | |
| A | | | * | : | J | Z | j | z | | | | | | | | |
| B | | | + | ; | K | [ | k | { | | | | | | | | |
| C | | | , | < | L | ¥ | l | \| | | | | | | | | |
| D | | | – | = | M | ] | m | } | | | | | | | | |
| E | | | . | > | N | ^ | n | → | | | | | | | | |
| F | | | / | ? | O | _ | o | ← | | | | | | | | |

**IMPORTANT**

Design the configuration of a system to provide an external protective or safety inter locking circuit for the PCs.

Under no circumstances will Mitsubishi Electric be liable or responsible for any consequential damage that may arise as a result of the installation or use of this equipment.

All examples and diagrams shown in this manual are intended only as an aid to understanding the text, not to guarantee operation. Mitsubishi Electric will accept no responsibility for actual use of the product based on these illustrative examples.

Owing to the very great variety in possible applications of this equipment, you must satisfy yourself as to its suitability for your specific application.

MITSUBISHI ELECTRIC CORPORATION

HEAD OFFICE: MITSUBISHI DENKI BLDG MARUNOUCHI TOKYO 100 TELEX: J24532 CABLE MELCO TOKYO
NAGOYA WORKS : 1-14 , YADA-MINAMI 5 , HIGASHI-KU , NAGOYA , JAPAN

When exported from Japan, this manual does not require application to the Ministry of International Trade and Industry for service transaction permission.

IB (NA) 66406-A (9303) MEE Printed in Japan

Specifications subject to change without notice.